

# Distributed Routing Software

---

## System Software Guide

Part Number: AA-QL2EC-TE

**January 1996**

This manual provides information about installing, configuring, and operating the Distributed Routing Software system software.

**Revision/Update Information:** This is a revised manual.

**Software Version:** Distributed Routing Software V1.1

Digital Equipment Corporation makes no representations that the use of its products in the manner described in this publication will not infringe on existing or future patent rights, nor do the descriptions contained in this publication imply the granting of licenses to make, use, or sell equipment or software in accordance with the description.

Possession, use, or copying of the software described in this publication is authorized only pursuant to a valid written license from Digital or an authorized sublicensor.

© Digital Equipment Corporation 1996  
All Rights Reserved.  
Printed in U.S.A.

The following are trademarks of Digital Equipment Corporation: DEC, DECnet, OpenVMS, PATHWORKS, ThinWire, VAX, VAXcluster, VMS, VT, and the DIGITAL logo.

The following are third-party trademarks:

AppleTalk is a registered trademark of Apple Computer, Inc.

Intel is a trademark of Intel Corporation.

IBM is a registered trademark of International Business Machines Corporation.

MS-DOS is a registered trademark of Microsoft Corporation.

NetWare and Novell are registered trademarks of Novell, Inc.

Proteon, ProNET, and TokenVIEW are registered trademarks of Proteon, Inc.

UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company Ltd.

This manual was produced by Shared Engineering Services.

---

# Contents

## Preface

### 1 Getting Started

Before You Begin .....	1-1
Using Router Consoles .....	1-1
Local Console .....	1-2
Remote Consoles .....	1-2
Telnet Connections .....	1-2
Remote Login Names and Passwords .....	1-2
Logging In Remotely .....	1-3
Using the Router User Interface .....	1-4
User Interface Processes in the Router .....	1-4
Using the OPCON process .....	1-6
Entering the OPCON Process .....	1-7
Process Prompts .....	1-7
Getting Help .....	1-7
Communicating with Different Processes .....	1-8
Connecting to a Process .....	1-8
OPCON Commands .....	1-8
Getting Back to OPCON .....	1-17
Accessing and Using the Configuration Process .....	1-17
Accessing Configuration Commands .....	1-18
Entering the CONFIG Process .....	1-19

Accessing an Interface Configuration Prompt .....	1-19
Displaying an Interface Configuration .....	1-20
Configuring Network Interfaces .....	1-20
Exiting an Interface Configuration Prompt .....	1-21
Accessing a Protocol Configuration Prompt .....	1-21
Exiting a Protocol Configuration Prompt .....	1-22
Restarting the Router .....	1-23
Accessing and Using the Console Process .....	1-23
Accessing the Console Process .....	1-23
Entering an Interface Console Prompt .....	1-24
Monitoring the Network Interface .....	1-25
Exiting an Interface Console Prompt .....	1-25
Entering a Protocol Console Prompt .....	1-25
Exiting a Protocol Console Prompt .....	1-26
Protocol Names and Numbers .....	1-26
Accessing Feature Configuration and Console Prompts .....	1-27
Accessing Feature Configuration .....	1-28
Accessing the Features Functions from the Console Process .....	1-28
System Security .....	1-29

## 2 Configuring the System

Using CONFIG .....	2-1
Quick Configuration .....	2-2
Configuring User Access .....	2-5
Tech Support Access .....	2-5
Entering and Exiting CONFIG .....	2-5
CONFIG Commands .....	2-6
CONFIG-ONLY Mode .....	2-29

## 3 Booting, Dumping, and Upgrading the System

Using Boot CONFIG .....	3-1
Boot Files .....	3-2

Entering and Exiting Boot CONFIG .....	3-2
Dumping the System Memory Contents .....	3-2
Configuring Dumping .....	3-3
Dump Files .....	3-4
Boot and Dump Directories .....	3-4
Displaying Dump Information .....	3-4
Boot CONFIG Commands .....	3-6
Trivial File Transfer Protocol (TFTP) .....	3-17
The BOOTP Forwarding Utility .....	3-23
The Bridging Router as a BOOTP Client .....	3-23
The Bridging Router as a BOOTP Relay Agent .....	3-23
BOOTP Server .....	3-24
Enabling/Disabling BOOTP Forwarding .....	3-24
Configuring a BOOTP Server .....	3-25
Upgrading the System Software .....	3-25
Starting the Upgrade .....	3-26
The reload Command .....	3-27
The load remote Command .....	3-27
IP Host-only Mode .....	3-27
Crash Handler .....	3-28

## 4 Monitoring the System

Using GWCON .....	4-1
Entering and Exiting GWCON .....	4-2
GWCON Commands .....	4-3

## 5 Using the Event Logging System

Using the Event Logging System (ELS) .....	5-1
Event Logging Concepts .....	5-2
Causes of Events .....	5-2
Interpreting a Message .....	5-3
Subsystem .....	5-3

Event Number .....	5-3
Message Text .....	5-4
Other ELS Terms .....	5-5
Logging Level .....	5-5
Groups .....	5-7
ELS Configuration Environment .....	5-8
ELS Console Environment .....	5-8
Entering and Exiting ELS .....	5-8
Entering and Exiting ELS Configuration Environment .....	5-9
Entering and Exiting ELS Console Environment .....	5-9
ELS Configuration Commands .....	5-10
Monitoring the Event Logging System .....	5-20
Process Characteristics .....	5-20
Using the ELS Monitor Process Effectively .....	5-20
Using ELS to Troubleshoot a Problem .....	5-21
Accessing the ELS Console Environment .....	5-23
ELS Console Commands .....	5-23

## A The MONITR Process

The MONITR Process .....	A-1
Commands Affecting MONITR .....	A-2
Entering and Exiting MONITR .....	A-3
Receiving MONITR Messages .....	A-3

## Index

## Figures

1-1 Router Software Processes .....	1-5
2-1 CONFIG in the Router Software Structure .....	2-2
2-2 Router Configuration Determination Flow Chart .....	2-4
4-1 Config in the Router Software Structure .....	4-2

5-1	ELS in the Router Software Structure .....	5-2
5-2	Message Generated by an Event .....	5-3
A-1	MONITR in the Router Software Structure .....	A-2

## Tables

1-1	OPCON Commands .....	1-9
1-2	Protocol Numbers and Names .....	1-27
1-3	Bridging Router Feature Numbers and Names .....	1-28
2-1	CONFIG Command Summary .....	2-7
2-2	Bridging Router Feature Numbers and Names .....	2-17
2-3	Protocol Numbers and Names .....	2-22
2-4	Default and Maximum Settings for Interfaces .....	2-26
3-1	Boot CONFIG Commands .....	3-6
4-1	GWCON Command Summary .....	4-4
4-2	Bridging Router Feature Numbers and Names .....	4-12
4-3	Protocol Numbers and Names .....	4-17
5-1	Packet Completion Codes (Error Codes) .....	5-4
5-2	Logging Levels .....	5-6
5-3	ELS Configuration Command Summary .....	5-11
5-4	ELS Console Command Summary .....	5-23



---

# Preface

## Objectives

This *System Software Guide* contains information for installing, configuring, and operating the Digital Distributed Routing Software. More specifically, this guide enables you to

- Configure, monitor, and use the MOS (Micro-Operating System) bridging router software.
- Configure, monitor, and use the interfaces and Link Layer software supported by your bridging router.

## Audience

This guide is intended for persons who install and operate computer networks. Although experience with computer networking hardware and software is helpful, you do not need programming experience to configure, monitor, and manage your network.

## Organization

This manual is organized as follows:

- Chapter 1 describes how to use router consoles to configure and control your network. Specifically, it discusses the Operator Console process, OPCON, and how you use it to configure and monitor protocols, enhance performance, and manage network interfaces.
- Chapter 2 describes how to access and use the various configuration commands.

- Chapter 3 describes how to use Boot CONFIG, Boot CONFIG Commands, BOOTP Forwarding, and Trivial File Transfer Protocol (TFTP). The information in this chapter is used to configure the module for a software upgrade or to save your configuration database to a remote system.
- Chapter 4 describes how to access and use the GWCON monitor process.
- Chapter 5 describes how to access and use the ELS event reporting system.
- Appendix A describes the MONITR process used as a report generator in conjunction with ELS.

## **Associated Digital Documents**

The following documents provide additional information about the router hardware and software:

- *Bridging Configuration Guide*, AA-QL29C-TE
- *Event Logging System Messages Guide*, AA-QL2AC-TE
- *Network Interface Operations Guide*, AA-QL2BC-TE
- *Routing Protocols Reference Guide*, AA-QL2CC-TE
- *Routing Protocols User's Guide*, AA-QL2DC-TE
- *System Network Architecture Guide*, AA-QU5SA-TE

## Conventions Used in This Guide

Special type	This special type in examples indicates system output or user input.
<b>Boldface</b>	Boldface type in examples indicates user input.
lowercase-italics	Lowercase italics in command syntax or examples indicate variables for which either the user or the system supplies a value.
[ ]	Brackets enclose operands or symbols that are either optional or conditional. Specify the operand and value if you want the condition to apply. Do not type the brackets in the line of code.
<span style="border: 1px solid black; padding: 2px;"><i>key</i></span>	Indicates that you press the specified key.
<span style="border: 1px solid black; padding: 2px;">Ctrl/x</span>	Indicates that you should hold the CONTROL key down and press the key specified by the x. The server displays the key combination as ^x.
<span style="border: 1px solid black; padding: 2px;">RET</span>	Indicates that you should press the Return key.
<u>underscore</u>	Characters underscored in a command listing represent the least number of characters you must enter to identify that command to the interpreter.



---

## Getting Started

This chapter discusses components related to the bridging router and the system software. It explains how to configure the bridging router software and how to monitor the software and network interfaces.

### Before You Begin

Before you start, refer to the following checklist to make sure that your router is installed as described in your hardware installation guide.

#### DID YOU...

- Install all necessary hardware? Refer to the appropriate hardware module installation guide.
- Connect the console terminal (video terminal) to the router? Refer to the appropriate hardware module installation guide.
- Connect your router to the network using the proper network interfaces and cables? Refer to the appropriate hardware module installation guide.
- Run all necessary hardware diagnostics? Refer to the appropriate hardware module installation guide.
- Install the router software if it was not pre-installed or if you are upgrading the software? Refer to your module's hardware installation manual or to the Release Notes shipped with the software upgrade.

### Using Router Consoles

The router console lets you use the router software user interface to monitor and change the function of the router's networking software. The router supports local and remote consoles. The following sections explain each type.

## Local Console

A local console is required for the initial configuration of the router software. The local console is directly connected to the router hardware or through a modem. The router software automatically handles console activity. When upgrading the software, you may need to use the local console.

**Note:** After the initial configuration, you do not need a local console for router management if either IP forwarding or TCP/IP Host Services is enabled.

## Remote Consoles

Remote consoles connect to the router using a standard remote terminal protocol and provide the same functionality as local consoles. No more than two remote consoles can be used at the same time on the same router. You can connect remote consoles to the router through a Telnet connection. You may disable the remote console feature.

## Telnet Connections

The router supports both the Telnet client and server. The remote console on the router acts as a Telnet server. The router acts as a Telnet client when connecting from the router to either another router or a host using the **telnet** command in the Operator Console (OPCON (\*)) process.

## Remote Login Names and Passwords

During a remote login, the router prompts you for a login name and password.

Use the **add user** command to supply a login name and password for remote access to the router. These user-configurable login names and passwords control access to the router.

**Note:** The router drops the Telnet connection if either of the following conditions occurs: you do not enter a login name and valid password within a minute of the initial prompt, or you enter three incorrect passwords in succession.

Multiple users with login permissions may also be added using the **add user** command. See Chapter 3 for more details on the **add user** command.

## Logging In Remotely

To log in from a remote system:

1. Connect to the router by starting Telnet and supplying the router's name or Internet Protocol address. For example:

```
% telnet brandenburg
```

*or*

```
% telnet 128.185.132.43
```

**Note:** To use router names, your network must have a name server.

2. At the login prompt, enter your login name. The prompts are shown below:

```
login:  
Password:
```

**Note:** The login and password prompts appear only if at least one administrative user account is set up on the router. Otherwise, logins are disabled by default.

**Note:** The login name is case sensitive.

3. If a password is not set, press **[RET]** at the password prompt.
4. Press **[Enter]** to obtain the main router prompt. This is displayed as an asterisk (\*). Depending on the state of the Remote Operator Console process (ROPCON), you may need to press **[RET]** more than once or press **[CTRL/p]** to obtain the asterisk (\*) prompt.

When the main prompt appears, you can enter commands from the keyboard. Use the **[BACKSPACE]** key to delete the last character entered; use **[DEL]** or **[CTRL/u]** to delete all the characters entered on the command line allowing you to re-enter a command.

5. To exit the router, return to the main router prompt (\*) and close the Telnet connection. For example:

```
IP Config> exit
Config> CTRL/p
* logout
%
```

You can use local telnet commands on your telnet client to close the Telnet connection.

## Using the Router User Interface

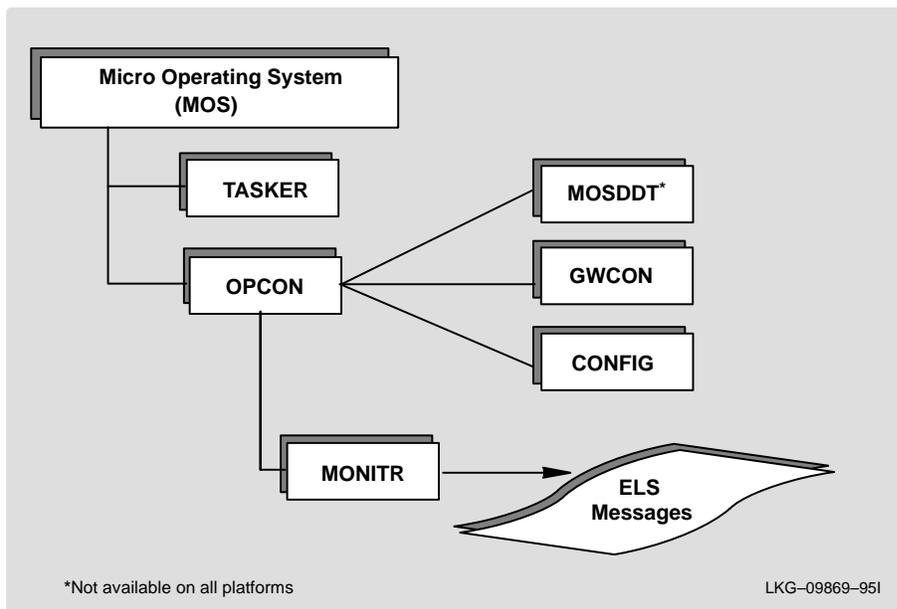
This section describes the router's user interface processes and explains how to communicate with them.

### User Interface Processes in the Router

The user interface to the router software consists of the main process, OPCON, and several subsidiary processes. These processes enable you to control and monitor the operations of the router. The three processes you communicate with most often are OPCON, Gateway Configuration (GWCON), and Configuration (CONFIG).

Figure 1–1 shows the processes and how they fit within the structure of the router software.

**Figure 1–1 Router Software Processes**



The following is a list of processes on the router:

- **OPCON** — Operates as the main operator control program.
- **COPCON** — Provides OPCON service for one directly-connected console.
- **ROPCON** — Provides OPCON service for two remotely-connected console terminals.
- **GWCON** (or **CGWCON**) — Provides online monitoring of the various software components, such as protocols, network interfaces, and event logging.
- **CONFIG** — Provides online control of various configuration parameters, such as network addresses and event logging. This manual concentrates on the CONFIG process and all its related protocol and interface configuration processes.

You can also run the **QUICK CONFIG** command from the CONFIG process. The router runs **QUICK CONFIG** if it determines that it has insufficient configuration information to enable it to run.

- MONITR — (Monitor) Performs message printing services.
- MOSDDT — The router debugging process.
- TASKER — Runs the router's main networking software and performs the router's internetwork data transfer operations.

The following modes are also accessible through the system:

- QUICK CONFIG — Provides a simple way to configure devices, set bridging and routing protocols, and create booting records. It is a special command of the CONFIG process.
- CONFIG-ONLY — Uses the same commands as those of the CONFIG process plus the **restart** and **reload** commands. It is a special CONFIG mode that is invoked when the router configuration is corrupted.

## Using the OPCON process

The Operator Console process (OPCON) is the root level process of the router software user interface. The main function of OPCON is to control the processes that are connected to consoles. Using OPCON commands, you can do the following:

- Configure a process.
- Change the intercept character.
- Reload the router software.
- Monitor a process.
- Communicate with other processes.
- Manipulate the output from a process.
- Restart the router software.
- Telnet to other routers or hosts.
- Display process statistics.
- Display information about router memory usage.

- Display status information about all router processes.

Refer to Figure 1–1 to see how OPCON fits into the router software structure.

When you access OPCON from a remote terminal, you access the ROPCON (Remote Operator Console) process. ROPCON and OPCON are functionally the same.

## Entering the OPCON Process

When you start the router for the first time (before users are configured), a platform-specific installation menu appears on the console followed by the OPCON prompt (\*). At this point, you can begin entering OPCON commands.

## Process Prompts

Each process uses a different prompt. You can tell which process your console is connected to by looking at the prompt. (If the prompt does not appear when you enter the **talk pid** command, press the `RET` key a few times.)

The following list shows the prompts for the three main processes:

<u>Process</u>	<u>Prompt</u>
OPCON	*
GWCON	+
CONFIG	Config>

At the prompt level, you can begin to enter commands from the keyboard. Use the `BACKSPACE` key to delete the last character typed in on the command line. Use `CTRL/u` to delete all the characters entered on the command line allowing you to re-enter a command.

## Getting Help

You can obtain help at any of the OPCON, GWCON, or CONFIG prompts by pressing `?` followed by `RET`. Help is displayed as a list of the commands available at that prompt level. Use **? (help)** to list the commands that are available from the current prompt level. You can enter a `?` after a specific command name to list its options.

## Communicating with Different Processes

The OPCON process handles the communication between users and the other router processes. If you are using a remote console, the name of the process handling your console is ROPCON instead of OPCON, but the operations are identical.

Processes interact with consoles by means of input and output channels. Using OPCON commands, you pass commands from your console to the input channel of any process and control any consoles attached to the output channel.

## Connecting to a Process

To connect your console to a process, use the OPCON **talk** *pid* command, where *pid* (process ID) is the number of the process to which you want to connect.

To find the pid number of a process, use the OPCON **status** command. The **status** command displays information about the router processes, such as the pids and process names.

## OPCON Commands

This section describes the OPCON commands. Each command includes a description, syntax requirements, and an example. The OPCON commands are summarized below. To use them, access the OPCON process and enter the OPCON command at the OPCON prompt (\*).

**Table 1–1 OPCON Commands**

<b>Command</b>	<b>Function</b>
<b>? (Help)</b>	Lists all the OPCON commands.
<b>Breakpoint</b>	Enters the DDT debugger.
<b>Divert</b>	Sends output from a process to a console or other terminal.
<b>Dump</b>	Dumps the contents of router memory to a remote host specified by the Boot config dump–entry records.
<b>Flush</b>	Discards the output from a process.
<b>Halt</b>	Stops a process output.
<b>Intercept</b>	Sets the OPCON default intercept character.
<b>Logout</b>	Logs out a remote console.
<b>Memory</b>	Reports the router’s memory usage.
<b>Restart</b>	Restarts (but does not reload) the router software.
<b>Reload</b>	Reloads the router software.
<b>Status</b>	Shows information about all router processes.
<b>Talk</b>	Contacts another router process and enables the use of its commands.
<b>Telnet</b>	Connects to a remote host.

**? (Help)**

List the commands that are available from the current prompt level. You can also enter a ? after a specific command name to list its options.

**Syntax:** ?

Example: ?

```
BREAKPOINT
DIVERT output from process
DUMP contents of memory
FLUSH output from process
HALT output from process
INTERCEPT character is
LOGOUT
MEMORY statistics
RESTART
RELOAD
STATUS of process(es)
TALK to process
TELNET to IP-Address (this terminal type)
```

## Breakpoint

Trap information in the DDT debugger, inspect memory, place breakpoints, or obtain a memory dump. Because the watchdog is always running, when you use the **breakpoint** command, the contents of memory are dumped when the watchdog timer expires.

**Note:** Do not use this command during normal operation because it halts operation of the software. If you accidentally enter the **breakpoint** command, quickly enter `[ESC/p]`.

**Syntax:** `breakpoint`

Example: `breakpoint`

## Divert

Send the output from a specified process to a specified terminal. This command allows you to divert the output of several processes to the same terminal to view the output simultaneously. The **divert** command is commonly used to redirect MONITR output messages to a specific terminal. The router allows only certain processes to be redirected.

After entering the command, enter the pid and tty# (number of the output terminal). The terminal number can be the number of either the local console (tty0) or one of the remote consoles (tty1, tty2,...ttn). The example below shows Event Logging System messages generated by the MONITR process (2) being sent to a remote console *tty1* (1).

**Syntax:** `divert pid tty#`

Example: `divert 2 1`

## Dump

Dumps the contents of router memory to a remote host and file name specified by the Boot config dump-entry records.

**Syntax:** `dump`

## Flush

Discard all subsequent output from a specified process until the **divert** or **talk** command is issued to that process. The router only allows certain processes to be redirected.

**Syntax:** `flush pid`

Example: `flush 2`

## Halt

Suspend all subsequent output from a specified process until the **divert**, **flush**, or **talk** command is issued to the process. The router cannot redirect all processes. **Halt** is the default state for output from a process.

**Syntax:** `halt pid`

Example: `halt 2`

## Intercept

Change the OPCON intercept character. You enter the **intercept** character from other processes to get back to the OPCON process. The default intercept character is `CTRL/p`.

**Note:** Do not set the intercept character to the return key or to a printable character.

**Syntax:** `intercept character`

Example: `intercept` CTRL/x

## Logout

Terminate remote console sessions.

**Syntax:** `logout`

Example: `logout`

## Memory

Obtain and display information about the router's global heap memory usage.

**Syntax:** `memory`

Example: `memory`

Number of bytes: Busy = 319544, Idle = 1936, Free = 1592

*Idle* Specifies the number of bytes previously allocated but freed and available for reuse. The sum of the Idle and Free memory equals the total available heap memory.

*Free* Specifies the number of bytes that were never allocated from the initial free storage area. The sum of the Idle and Free memory equals the total available heap memory.

## Restart

Reinitialize the software. When the software initializes, it forces a bus reset, which causes the connected network interfaces to perform their self-test routines, to clear all routing tables, and to drop any packets in the router. Before the restart takes effect, the software prompts you for confirmation. This command is functionally equivalent to pressing the reset button, except that in the latter case the router dumps (if dumping is enabled). The router does not dump when it executes the **restart** command. A restart reinitializes the router, and runs the same image.

**Note:** If you use this command from a remote console, your Telnet session will be lost.

**Syntax:** restart

Example: **restart**

```
Are you sure you want to restart the gateway (Yes or No)?
```

## Reload

Obtain a new copy of the router software specified by the boot-entry records. This new image is stored in FLASH and is run each time the router boots. Before the reload takes effect, the software prompts you for confirmation.

**Syntax:** reload

Example: **reload**

```
Are you sure you want to reload the gateway (Yes or No)?
```

## Status

Display information about all router processes. By entering the pid after the **status** command, you can select the status of a particular process. The example below shows the total status display. The actual processes may be different depending on the hardware you are using.

**Syntax:** status *pid*

Example: **status**

MOS Operator Control

```
*status
Pid  Name      Status TTY  Comments
1    COpCon    IOW   TTY0
2    Monitr    DET   --
3    Tasker    RDY   --
4    MOSDDT    DET   --
5    CGWCon    IOW   --
6    Config    DET   --
7    ROpCon    IDL   TTY1
8    ROpCon    RDY   TTY2 16.24.10.98
*
```

*Pid* The process ID. This is the process that communicates with OPCON, or it can be an argument to the STATUS command to request status information about a specific process.

*Name* The process name.

*Status* One of the following:

- **IDL** – This process is idle and waiting for completion of an external event, such as asynchronous I/O.
- **RDY** – This process is ready to run and is waiting to use the CPU.
- **IOW** – This process is waiting for synchronous I/O to complete.
- **DET** – This process has output ready to be displayed and is either waiting to be attached to a display console or waiting to have its output diverted to a specified console.
- **FZN** – This process is “frozen” due to an error. This usually means the process is trying to use a device which is faulty or incorrectly configured.

<i>TTYn</i>	The output terminal, if any, to which the process is currently connected. TTY0 indicates a local console while TTY1 or TTY2 denote Telnet consoles. “Sink” indicates that a process was flushed. Two dashes (- -) indicate that a process was halted.
<i>Comments</i>	The user’s login IP address provided when a user is logged on using telnet (ROpCon).

## Talk

Attach to other router processes, such as GWCON or CONFIG. After connecting to a new process, you can send specific commands to and receive output from that process. You cannot talk to the TASKER or OPCON processes. To obtain the pid, use the OPCON **status** command.

**Syntax:** `talk pid`

Example: `talk 5`

## Telnet

Remotely attach to another bridging router or to a remote host (*ip address*). The only optional parameter is the terminal type that you want to emulate. Enter any 80 character terminal type.

A router has a maximum of five telnet sessions. Two servers (inbound to the router), and three clients (outbound from the router).

Telnet uses the terminal type that you supply on the command line on the remote host to define the characteristics of the telnet session.

**Note:** To use telnet in a pure bridging environment, you must enable Host Services.

**Syntax:** `telnet ip address <terminal type>`

Example: `telnet 128.185.10.30`  
or  
`telnet 128.185.10.30 vt100`

```
Trying 128.185.10.30 ...
Connected to 128.185.10.30
Escape character is '^']'
```

If you are telneting to a non-existent IP address, the router displays the following:

```
Trying 128.185.10.30 ...
```

To enter the telnet command mode, type the escape character `^]`.

```
telnet>
```

The **telnet** command mode consists of four sub-commands that allow you to close the current connection, to quit the current session, to send an ayt (Are You There) to your current connection, or to display status about the connection. For example:

```
telnet> ?
close          close current connection
quit          exit telnet
status        print status information
send          transmit special characters ('send ?' for more)
?            print this help information
```

**Note:** Press `RET` to resume a telnet session that you temporarily escaped (by pressing `Ctrl-]`).

The **status** and **send** sub-commands elicit one of two responses depending on whether or not the user is connected to another host. For example:

Connected to a host:

```
telnet> status
Connected to 128.185.10.30
Operating in character-at-a-time mode.
Escape character is ^].
telnet> send ayt
```

**Note:** The **send** command currently only supports ayt.

Not connected to a host:

```
telnet> status
Need to be connected first.
telnet> send ayt
Need to be connected first.
```

Use the **close** sub-command to close a connection to a remote host and terminate the telnet session. Use the **quit** sub-command to exit **telnet** command mode, to close a connection, or to terminate a telnet session.

```
telnet> close

or

telnet> quit

logout
*
```

## Getting Back to OPCON

You must always return to OPCON before you can communicate with another process. To get back to OPCON, press **[CTRL/p]**. For example, if you are connected to the GWCON process and you want to connect to the CONFIG process, you must press **[CTRL/p]** to return to OPCON first. The **[CTRL/p]** key combination is called the intercept character.

## Accessing and Using the Configuration Process

This section describes how to access the configuration process.

All interfaces described in the *Network Interface Operations Guide* and all protocols described in the *Routing Protocols User's Guide* have commands that are executed by accessing the *configuration* process to configure, enable, or modify the interface or protocol.

The procedures to access this process are basically the same for both. The following sections describe these procedures.

## Accessing Configuration Commands

You access configuration commands through the router's CONFIG process.

When you log onto the router from a directly connected console, your process is the Connected Operator Console process (COPCON) and its process identifier (pid) 1. When you log onto the router through a telnet session your process is one of the two Remote Operator Console processes (ROPCON) with pid 7 or 8. These processes are collectively known as OPCON.

As one of the OPCON processes, you can connect to (**talk**) any of the four user-accessible processes:

- MONITR – The Monitor process (pid 2) that displays ELS messages.
- GWCON – The Gateway Console process (pid 5) that displays router statistics and lets you manipulate the volatile environment.
- CONFIG – The Configuration process (pid 6) that configures the nonvolatile environment.
- DEBUG – The Debug process (pid 4) that is used for debugging.

Only one user can **talk** to each of these processes at a time. If another user chooses to **talk** to the same process you are currently using, the message “Output Redirected” is displayed, and you are returned to your originating process. The user who redirected the process that you were talking to receives all redirected output from the process that had not yet appeared on your screen. Any command executing while you were talking to that process completes, unless the current user cancels it.

You use the Configuration process (CONFIG) to manipulate configuration information for the router. Within the context of the CONFIG process, and available from the `Config>` prompt, are configuration commands that are logically grouped according to interface or protocol function. Each interface and protocol configuration module has its own prompt. For example, the prompt for the TCP/IP protocol command interface is `IP config>`.

To access a configuration prompt, do the following:

- Enter the CONFIG command process from OPCON and obtain the CONFIG prompt: `Config>`

- Invoke the desired interface or protocol configuration prompt from the CONFIG prompt using the appropriate command described in the sections that immediately follow. You can enter the minimum number of characters that makes the command unique such as **p**, **pr**, and so forth for **protocol**, or **n**, **net**, and so forth for **network**.

The following sections describe these procedures in more detail.

## Entering the CONFIG Process

To enter the CONFIG command process from OPCODE and obtain the CONFIG prompt (Config>), perform the following steps:

1. At the OPCODE prompt, enter the **status** command to find the pid of CONFIG.

```
* status
Pid      Name      Status  TTY    Comments
  1      COpCon    IOW     TTY0
  2      Monitr    DET     --
  3      Tasker    IDL     --
  4      MOSDDT    DET     --
  5      CGWCon    IOW     --
  6      Config    IOW     --
  7      ROpCon    IOW     TTY1    128.186.314.132
  8      ROpCon    RDY     TTY2    128.186.314.133
```

2. Enter the OPCODE **talk** command and the pid for CONFIG. The pid for CONFIG is 6.

```
* talk 6
```

The console displays the CONFIG prompt (Config>). If the prompt does not appear, press **RET** again.

## Accessing an Interface Configuration Prompt

This process gives you access to a specific interface's configuration commands. To access the router's configuration process, perform the following steps:

1. At the CONFIG prompt, enter the **list devices** command to see the network interface numbers for which the router is currently configured. For example:

The following example applies to the DECswitch 900EE.

```
Config>list devices

Ifc 0 (Ethernet): CSR      0, vector 0
Ifc 1 (Ethernet): CSR      1, vector 0
Ifc 2 (Ethernet): CSR      2, vector 0
Ifc 3 (Ethernet): CSR      3, vector 0
Ifc 4 (Ethernet): CSR      4, vector 0
Ifc 5 (Ethernet): CSR      5, vector 0
```

The following example applies to the RouteAbout Access EW.

```
Config>list devices

Ifc 0 (Ethernet): CSR 1001600, CSR2 1000C00, vector 94
Ifc 1 (WAN PPP): CSR 1001620, CSR2 1000D00, vector 93
Ifc 2 (WAN PPP): CSR 1001640, CSR2 1000E00, vector 92
```

2. Record the desired interface numbers.
3. Enter the CONFIG **network** command and the number of the interface you want to configure. For example:

```
Config> network 1
```

The appropriate configuration prompt (such as TKR Config> for token-ring), now appears on the console.

**Note:** Not all network interfaces are user-configurable. For interfaces that cannot be configured, you receive the following message:

```
That network is not configurable.
```

## Displaying an Interface Configuration

From some interface configuration prompts, you might list configuration information specific to that selected interface by using the **list** command. For example:

```
TKR Config> list

Packet size (INFO field):      4472
Speed:                          16 Mb/sec
Media:                           Unshielded
```

## Configuring Network Interfaces

Refer to the specific chapters in the *Network Interface Operations Guide* for complete information on configuring your types of network interfaces. The possible network interfaces are Token-ring, Ethernet, FDDI, ISDN, and Serial Line.

## Exiting an Interface Configuration Prompt

After you configure the desired interface information, exit the interface configuration prompt by performing the following steps:

1. Return to the CONFIG process by entering the **exit** command. For example:

```
TKR Config> exit
```

2. Return to the OPCON process by entering the OPCON intercept character. For example:

```
Config> CTRL/p
```

## Accessing a Protocol Configuration Prompt

**Note:** To list all of the available protocols, type **p ?** from the `Config>` prompt. You can also use the **list config** command to discover this same information

To access the desired protocol configuration prompt from the CONFIG prompt, perform the following steps:

1. At the CONFIG prompt, enter the **list configuration** command to see the numbers and names of the protocols contained in the software. For example:

```
Config>list conf
Hostname: wg07

Maximum packet size: [autoconfigured]
Maximum number of global buffers: [autoconfigured]
Number of Restarts before a Reload/Dump: 64
Logging disposition: detached
Console inactivity timer (minutes): 0
Physical console login: disabled
Modem control: disabled
Contact person for this node: [none]
Location of this node: [none]

Configurable Protocols:
Num Name Protocol
0 IP DOD-IP
3 ARP Address Resolution
4 DN DNA Phase IV
5 XNS Xerox Network Systems*
6 VIN Banyan Vines*
7 IPX NetWare IPX
8 OSI ISO CLNP/ISIS/ISIS
9 DVMRP Distance Vector Multicast Routing Protocol
```

```
10 BGP   Border Gateway Protocol
11 SNMP  Simple Network Management Protocol
12 OSPF  Open SPF-Based Routing Protocol
14 APL   AppleTalk
15 DDS   Apollo Domain*
22 AP2   AppleTalk Phase 2
23 ASRT  Adaptive Source Routing Transparent Enhanced Bridge
24 HST   TCP/IP Host Services

7857 bytes of configuration memory free
```

\* Not supported in this release.

2. From the CONFIG prompt, enter the **protocol** command with the number or *short name* of the protocol you want to configure. You can obtain the protocol number and *short name* from the **list configuration** command display. The following example shows the command to access the IP protocol configuration prompt by the protocol *short name*:

```
Config> protocol IP
```

The protocol configuration prompt then appears on the console. This example shows the IP protocol prompt `IP config>`:

```
IP config>
```

You can obtain the same result by entering the **protocol** command followed by the protocol “number.” The following example shows the command you use to access the IP protocol configuration prompt by the protocol number:

```
Config> protocol 0
```

The protocol configuration prompt then appears on the console.

```
IP config>
```

You can now enter that protocol’s configuration commands. See the corresponding protocol section of the *Routing Protocols User’s Guide* for more information on specific protocol configuration commands.

## Exiting a Protocol Configuration Prompt

After completing the desired changes to configuration information for a protocol, exit to the `Config>` prompt by performing the following steps:

1. To exit from protocol configuration, enter the protocol **exit** command. For example:

```
IP config> exit
```

2. Return to the OPCON process by entering the OPCON intercept character (`CTRL/p`). For example:

```
Config> CTRL/p
```

## Restarting the Router

Changes that you make to the protocol parameters through CONFIG, except for certain **NCP SET** commands, do not take effect until you restart the router.

**Note:** The changes you make through CONFIG are retained in a configuration database in non-volatile memory. They are retained during power downs and are recalled when you restart the router.

To restart the router, enter the OPCON **restart** command. For example:

```
* restart  
Are you sure you want to restart the router? (Yes or No)[No]: yes
```

## Accessing and Using the Console Process

All interfaces described in the *Network Interface Operations Guide* and all protocols described in the *Routing Protocols User's Guide* have commands that are executed by accessing the *console* process to monitor network statistics, test network interfaces, view the configured protocols on the router, or make temporary changes to the volatile configuration of the router.

You access the console process through the GWCON process. This process is described in detail in Chapter 4 of this manual. The procedures to access this process are basically the same for both. The following sections describe these procedures.

### Accessing the Console Process

To monitor information related to a specific interface or protocols, access the interface *console* process by performing the following steps:

1. Enter the **status** command to find the pid of GWCON. For example:

```
* status
```

Pid	Name	Status	TTY	Comments
1	COpCon	IOW	TTY0	
2	Monitr	DET	--	
3	Tasker	IDL	--	
4	MOSDDT	DET	--	
5	CGWCon	IOW	--	
6	Config	IOW	--	
7	ROpCon	IOW	TTY1	janb
8	ROpCon	RDY	TTY2	

\*

2. At the OPCON prompt, enter the OPCON **talk** command and the pid number for GWCON. For example:

```
* talk 5
```

3. The GWCON prompt (+) is displayed on the console. If the prompt does not appear when you first enter GWCON, press **RET** again.

### Entering an Interface Console Prompt

To access an interface console prompt from the GWCON prompt, perform the following steps:

1. At the GWCON prompt, enter the **configuration** command to display the protocols and networks for which the router is configured. For example:

```
+ configuration
Num Name Protocol
0 IP DOD-IP
3 ARP Address Resolution
7 IPX NetWare IPX
11 SNMP Simple Network Management Protocol
12 OSPF Open SPF-Based Routing Protocol
23 ASRT Source Routing Transparent Enhanced Bridge

Num Name Feature
1 BRS Bandwidth Reservation

4 Networks:
Net Interface MAC/Data-Link Hardware State
0 Eth/0 Ethernet/IEEE 802.3 Ethernet/802.3 Up
```

2. Enter the GWCON **network** command and the number of the interface you want to monitor. For example:

```
+ network 3
```

In this example the X.25 console prompt appears on the console. You can then view information about the X.25 interface by entering the X.25 console commands.

x.25>

## Monitoring the Network Interface

Refer to the specific chapters in the *Network Interface Operations Guide* for complete information about monitoring your particular network interfaces.

## Exiting an Interface Console Prompt

To exit the interface console prompt and return to the OPCON process, do the following:

1. Enter the **exit** command to return to the GWCON process. For example:

```
x.25> exit
```

2. Press **CTRL/p** to return to the OPCON process. For example:

```
+ CTRL/p
```

## Entering a Protocol Console Prompt

To enter a protocol console prompt from the GWCON prompt, perform the following steps:

1. At the GWCON prompt (see Chapter 4 for information about accessing the GWCON prompt), enter the **configuration** command to see the protocols and networks configured for the router. For example:

```
+ configuration
4 Protocols:
Num Name Protocol
0 IP DOD-IP
3 ARP Address Resolution
11 SNMP Simple Network Management Protocol
12 OSPF Open SPF-Based Routing Protocol
```

2. Enter the GWCON **protocol** command with the protocol number or *short name* of the desired protocol displayed in the configuration information. In the following example, the command has been entered to access the IP protocol console process.

```
+ protocol 0
```

OR

```
+ protocol IP
```

The protocol console prompt then appears on the console. This example shows the TCP/IP prompt (IP>):

```
IP>
```

You can enter that protocol's console commands. See the corresponding protocol section of the *Routing Protocols User's Guide* for more information on specific protocol console commands.

### Exiting a Protocol Console Prompt

To exit the protocol console process and return to the OPCODE process, perform the following steps:

1. Return to the GWCON process by entering the protocol **exit** command. For example:

```
IP> exit
```

2. Return to the OPCODE process by entering the OPCODE intercept character (`CTRL/p`). For example:

```
+ CTRL/p
```

### Protocol Names and Numbers

Table 1–2 lists the numbers that you enter along with the **protocol** command when accessing a specific protocol configuration or console process.

**Note:** Not all protocols are supported by all hardware platforms.

**Table 1–2 Protocol Numbers and Names**

<b>Protocol Number</b>	<b>Protocol Short Name</b>	<b>Accesses the following protocol process</b>
0	IP	Internet Protocol
3	ARP	Address Resolution Protocol
4	DN	DNA – a subset of Network Control Program
7	IPX	Novell NetWare Internetwork Packet Exchange
8	OSI	ISO Open Systems Interconnect Connectionless Network Layer Protocol / ESIS / ISIS
11	SNMP	Simple Network Management Protocol
12	OSPF	Open Shortest Path First
14	APL	AppleTalk Phase 1
20	SDLC	SDLC Relay
22	AP2	AppleTalk Phase 2
23	ASRT	Adaptive Source Routing Transparent Bridge
24	HST	TCP/IP Host Services

## Accessing Feature Configuration and Console Prompts

This section describes how to access the bridging router feature configuration and console processes.

You run all bridging router features described in this manual by performing one of the following:

- Accessing the *configuration* process to configure, enable, or modify the feature.
- Accessing the *console* process to monitor information about each feature or to make temporary configuration changes.

The procedure to access these processes is basically the same for all features. The next sections describe these procedures.

## Accessing Feature Configuration

To access configuration commands for specific bridging router features that are not part of the protocol and network interface configuration processes, use the **feature** command of the CONFIG process.

To obtain a listing of the features available for your software release, enter `?` after the **feature** command. For example:

```
config> feature ?
BRS
Feature name or number [1] ?
```

To access that feature's configuration prompt, enter the **feature** command followed by the feature number or the *short name*. For example:

```
config> feature brs
Bandwidth reservation user configuration
BRS>
```

**Note:** For the Co-resident SNMP Agent feature, CONFIG also prompts you to enter the number of the interface that you want to configure or monitor.

Table 1–3 lists available bridging router feature numbers and names.

**Note:** Not all features are supported by all hardware platforms.

**Table 1–3 Bridging Router Feature Numbers and Names**

Feature Number	Feature Short Name	Accesses the following feature configuration and console process
0	WRS	WAN Restoral
1	BRS	Bandwidth Reservation
2	MCF	MAC Filtering

After you access the configuration prompt for that feature, you may enter specific configuration commands for that feature. To return to the CONFIG prompt, enter the **exit** command at the feature's configuration prompt.

## Accessing the Features Functions from the Console Process

Use the **feature** command from the GWCON process to access console commands for specific bridging router features that are not part of the protocol and network interface console processes.

Enter **?** after the **feature** command to obtain a listing of the features available for your software release. For example:

```
+ feature ?
```

To access that feature's console prompt, enter the **feature** command followed by the feature number or *short name*. For example:

```
+ feature brs
```

After you access the console prompt for that feature, you may enter specific console commands for that feature. To return to the GWCON prompt, enter the **exit** command at the feature's console prompt.

**Note:** For the Co-resident SNMP Agent feature, GWCON also prompts you to enter the number of the interface that you want to configure or monitor.

## System Security

You can add multiple users, each having login permissions, by using the **add user** command. See the section "Configuring User Access" in Chapter 2 for details about security issues and for information about the **add user** command.



---

## Configuring the System

This chapter describes the Configuration (CONFIG) process. It explains how you use CONFIG and Quick Config, and discusses the CONFIG commands.

### Using CONFIG

CONFIG is a process of the router user interface. Using CONFIG commands, you can do the following:

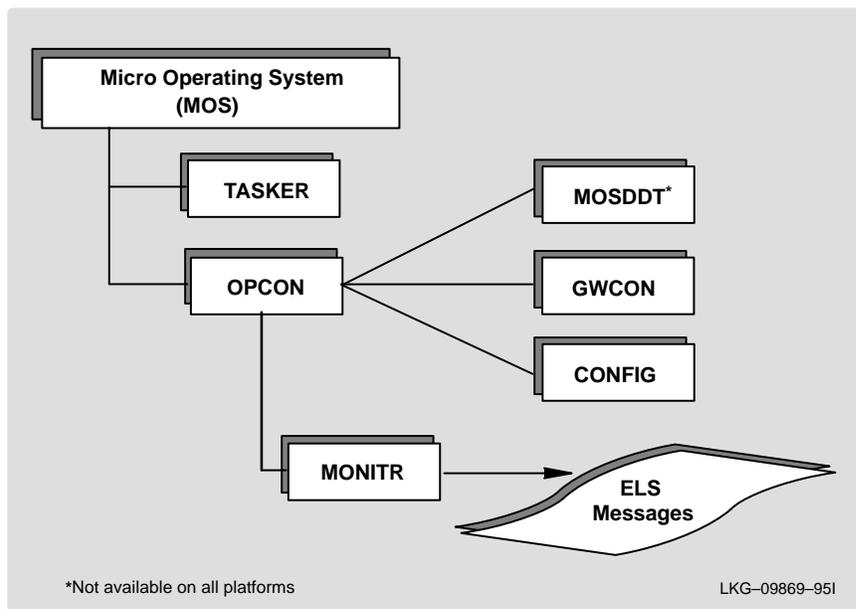
- Add an interface to or delete an interface from the non-volatile RAM (NVRAM) hardware configuration.
- Enter the Boot CONFIG command.
- Enter Quick Configuration (QUICK CONFIG).
- Clear, list, or update configuration information.
- Enable or disable console login and modem control.
- Communicate with protocol environments.
- Set or change various configuration parameters.

CONFIG lets you display or change the configuration information stored in the router's NVRAM. Changes to system and protocol parameters, except changes made by the NCP **set** command, do not take effect until you restart the router. (For more information, refer to the OPCON **restart** command.)

**Note:** The changes you make through CONFIG are retained in a configuration database in NVRAM. They are stored in FLASH memory and are recalled when you restart the router.

Figure 2–1 shows how CONFIG fits into the router software structure.

**Figure 2–1 CONFIG in the Router Software Structure**



The CONFIG interface contains commands that let you configure and monitor interfaces and protocols. Each command has its own prompt. For example, the prompt for the TCP/IP protocol is `IP config>`.

If you want to know the process and command you are communicating with, press `RET` to display the prompt. Some commands in this chapter, such as the **network** and **protocol** commands, allow you to access and exit the various levels in CONFIG.

## Quick Configuration

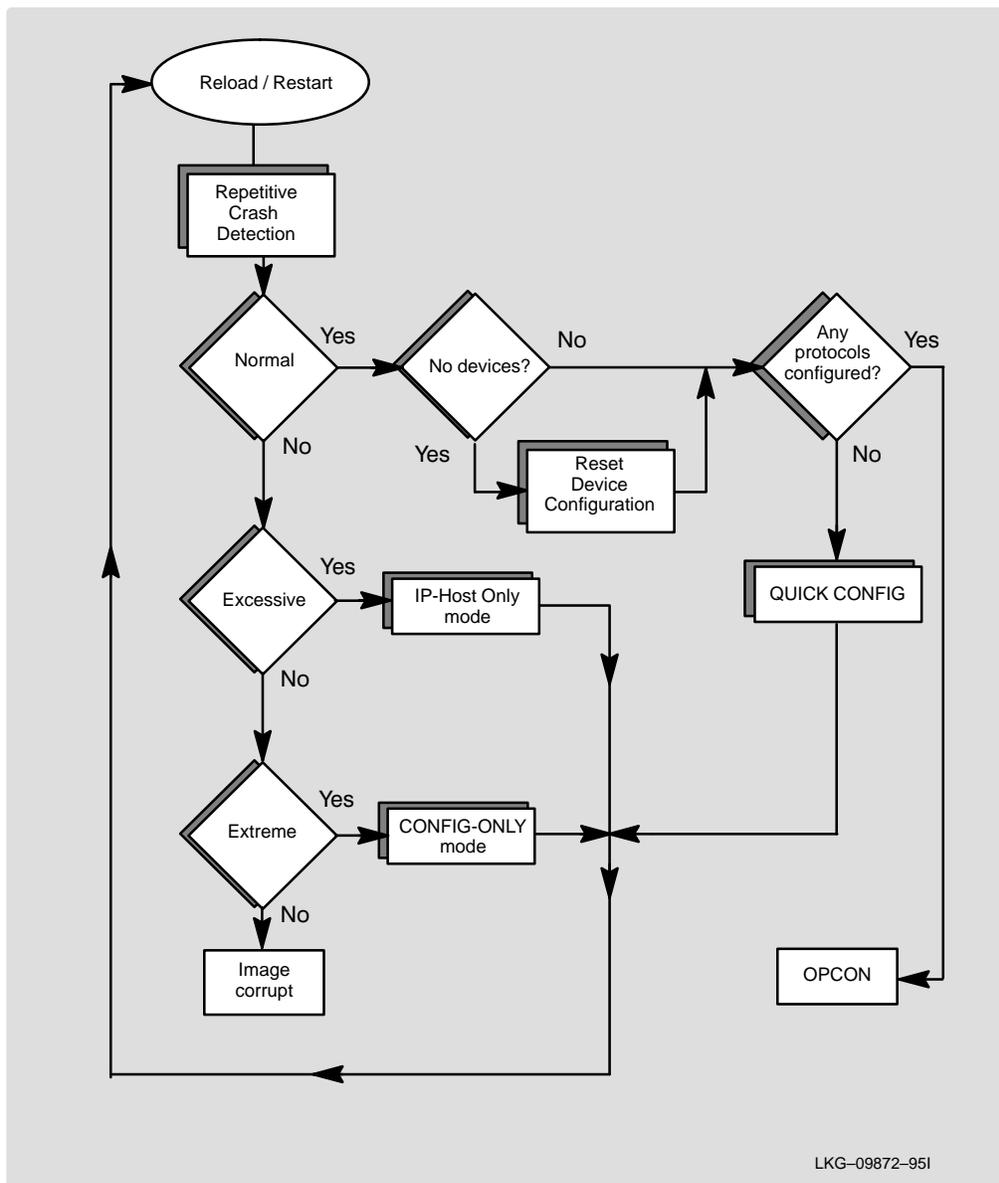
When you load, start, or restart the router with no configuration, you enter Quick Configuration (QUICK CONFIG).

Quick Configuration (QUICK CONFIG) allows you to configure a minimal set of parameters in order to boot the router and run it with a minimal configuration or to download a complete configuration by means of TFTP. QUICK CONFIG steps you through most user protocols for your hardware specific platform.

QUICK CONFIG enables you to configure the minimum number of parameters for different devices, bridging protocols, routing protocols, and booting records without exiting and entering the different configuration submenus of the CONFIG processes. For more information about QUICK CONFIG and to see how it is used, refer to the INSTALLATION MENU section in the installation manual that accompanies your router.

Figure 2–2 is a flow diagram that illustrates how the router determines which configuration you use.

Figure 2–2 Router Configuration Determination Flow Chart



## Configuring User Access

The router configuration permits user names (case sensitive, 8 characters maximum), passwords (8 characters maximum), and permission levels. Each user added must have a password and permission level. There are three permission levels: *Administrative*, *Operations*, and *Monitor*.

Users with *Administrative* permission can access any router function, including router configuration and user administration.

Users with *Operations* permission can view any network configuration parameter or statistic, run potentially disruptive tests, dynamically change the router operation, and restart the router.

Users with *Monitor* permission can only view configuration parameters or network statistics.

**Note:** Only a user with *Administrative* permission can change the configuration in NVRAM. In your hardware installation guide, NVRAM is referred to a non-volatile memory.

You receive the following prompts each time you log in on the console.

```
login:  
password:
```

## Tech Support Access

When system administrators add new users for the first time, they are asked if they want to add tech support access. If they answer yes, tech support access is granted *Administrative* access.

The software automatically selects the password, known to Customer Service, for this account. This password can be changed.

## Entering and Exiting CONFIG

To enter the CONFIG command environment from OPCON and obtain the CONFIG prompt perform the following steps:

1. At the OPCODE prompt, enter the **status** command to find the pid of CONFIG:

```
* s
```

Pid	Name	Status	TTY	Comments
1	COpCon	IOW	TTY0	
2	Monitr	DET	--	
3	Tasker	IDL	--	
4	MOSDDT	DET	--	
5	CGWCon	IOW	--	
6	Config	IOW	--	
7	ROpCon	IOW	TTY1	janb
8	ROpCon	RDY	TTY2	

2. Enter the OPCODE **talk** command and the pid for CONFIG:

```
* talk 6
```

The console displays the CONFIG prompt (Config>). Now you can enter CONFIG commands. If the prompt does not appear, press **RET** again. To exit CONFIG and return to OPCODE, enter the OPCODE intercept character **CTRL/p**.

## CONFIG Commands

This section describes each of the CONFIG commands. Each command includes a description, syntax requirements, and an example. The CONFIG commands are summarized in Table 2-1.

After accessing the CONFIG environment, enter the configuration commands at the Config> prompt.

**Table 2–1 CONFIG Command Summary**

<b>Command</b>	<b>Function</b>
<b>? (Help)</b>	Lists the CONFIG commands or lists the options associated with specific commands.
<b>Add</b>	Adds a user to the router or adds a dial-circuit, PPP-FR pseudo device or null device.
<b>Boot</b>	Enters Boot CONFIG command.
<b>Change</b>	Changes a current user's password or dial-circuit, PPP-FR pseudo device or null device.
<b>Clear</b>	Clears configuration information.
<b>Delete</b>	Deletes a configured user or a dial-circuit, PPP-FR pseudo device or null device.
<b>Disable</b>	Disables login from a remote console, disables remote access using a modem, or disables a specified interface.
<b>Enable</b>	Enables login from a remote console, enables remote access using a modem, or enables a specified interface.
<b>Event</b>	Enters the Event Logging System configuration environment.
<b>Feature</b>	Provides access to configuration commands for independent router features outside the usual protocol and network interface configuration.
<b>List</b>	Displays the system parameters, hardware configuration, users, obm ip-address, or v25-bis-address.
<b>Network</b>	Enters the configuration environment of the specified network.
<b>Patch</b>	Modifies the router's global configuration.
<b>Protocol</b>	Enters the command environment of the specified protocol.
<b>Qconfig</b>	Initiates Quick Config.
<b>Set</b>	Sets system wide parameters for buffers, hostname, inactivity timer, logging disposition and level, packet size, restart count, location, obm ip-address, and contact-person.
<b>Time</b>	Keeps track of system time and displays it on the console.
<b>Unpatch</b>	Restores variables of the patches to default values.
<b>Update</b>	Updates the current version of the configuration.

## ? (Help)

Lists the commands that are available from the current prompt level. You can also enter a ? after a specific command name to list its options.

**Syntax:** ?

Example: ?

```
ADD
BOOT
CHANGE
CLEAR
DELETE
DISABLE
ENABLE
EVENT
FEATURE
LIST
NETWORK
PATCH
PROTOCOL
QCONFIG
SET
TIME OF DAY PARAMS
UNPATCH
UPDATE
```

Example: **add** ?

```
device
user
v25-bis-address
```

## Add

Adds a device or user-access to the configuration.

**Note:** Hardware device records are automatically recreated (if they are missing) when the router restarts.

**Syntax:**    add        device . . .  
                          user . . .  
                          v25-bis-address . . .  
                          ISDN-address . . .

### **device** *device\_type additional-config-info*

The interface device (*device\_type*). CONFIG prompts you for any additional configuration parameters. This additional information varies by device and platform. If you know the additional configuration information that CONFIG prompts for, you can enter it on the command line. To obtain this information, refer to the site specification sheet shipped with the router.

**Note:** If you are adding more than one interface, the order in which you add them is important because the router assigns a sequential interface number to the device when it is added. This interface number is an index number in the device list. It links the device with other protocol configuration information, such as the IP addresses associated with the device. (For more information, refer to the **list devices** command.) All device and protocol configuration information related to network interfaces is stored by interface number. Any changes made to interface numbers invalidate some of the device configuration information in the protocols.

Example: **add device dial-circuit**

The following interfaces are not all available on all hardware platforms. Refer to your hardware installation guide to determine which interfaces are supported by your hardware platform.

Short Name	Interface Described
NULL	No device
PPP-FR	PPP over Frame Relay pseudo device
DIAL-CIRCUIT	

#### **user** *user\_name*

Adds a user access name to the router configuration. Each user added must be assigned one of three permission levels: *Administrative*, *Operations*, and *Monitor*. *Administrative* permission allows the user access to any router function, and allows the user to write to NVRAM. *Operations* permission allows a user to view any network configuration parameter or statistic, and run potentially disruptive tests, and restart the router. *Monitor* permission only allows the user to view the configuration parameters or network statistics.

When the first user is added, console login is automatically enabled. The user name is limited to 8 characters, and the password is limited to 8 characters.

**Note:** To add a user, you must have *Administrative* permission. All user names and passwords are case-sensitive.

Example: **add user John**

```
Enter password:
Enter password again:
Enter permission (A)dmin, (O)perations, (M)onitor [A]?
```

*Enter password*      The access password for the user.. Limited to 8 characters.

*Enter password again*      Confirms the access password for the user.

*Enter permission*      The access permission for the user: A, O, or M.

## **Boot**

Enter the Boot CONFIG command environment. For Boot CONFIG information, refer to Chapter 3.

**Syntax:**    boot

Example: **boot**

```
TFTP Boot/dump configuration
Boot config>
```

## Change

Modify an interface in the configuration, change your own password, or change user information.

**Syntax:** `change password`  
`user`

### password

Modifies the current user's password.

**Note:** To change a user password, you must have *Administrative* permission.

Example: **change password**

```
Enter current password:
Enter new password:
Enter new password again:
```

*Enter current password*      Your current password.

*Enter new password*          Your new password.

*Enter new password again*    Confirms your new access password. If your confirmation does not match the previous new password, the old password remains in affect.

### user

Modifies the user information that was previously configured with the **add user** command.

**Note:** To change a user, you must have *Administrative* permission.

Example: **change user**

```
User name: []?
change password? (Yes or No)?
Enter permission: (A)dmin, (O)perations, or (M)onitor [A]?
```

## Clear

Delete the router's configuration information from NVRAM.

To clear a protocol from NVRAM, enter the **clear** command and the protocol name. To clear all information from the NVRAM except for device information, use the **clear all** command. If you want to clear all information including the device information, use the **clear all** and **clear device** commands.

**Note:** To clear user information, you must have *Administrative* permission.

**Caution:** Only use this command after calling your Customer Service representative.

**Syntax:**    clear     all  
                  ap2 (AppleTalk 2)  
                  apl (AppleTalk 1)  
                  arp (ARP)  
                  asrt (Adaptive Source Route Protocol)  
                  boot information  
                  device information  
                  dialer-circuit  
                  dls (Data Link Switching)  
                  dn (DNA IV)  
                  dvm (Distance Vector Multicast Routing Protocol)  
                  els (Event Logging System Information)  
                  fr (Frame Relay)  
                  ip (IP)  
                  ipx (Novell IPX)  
                  isdn  
                  osi (OSI)  
                  ospf (OSPF routing protocol)  
                  ppp (Point-to-Point)

slc (Serial Line Controller SRAM block)  
snmp  
srb (Source Route Bridge)  
srly (SDLC Relay)  
stb (Spanning Tree Bridge)  
tcp/ip-host  
time (Time of day information)  
user info (User Access information)  
x25

Example: `clear all`

```
You are about to clear all non-device configuration information
Are you sure you want to do this? [Yes or (No)]:
```

**Note:** This message appears whenever you erase any configuration parameter. The **clear user** command clears all user information except the router console login information. The value is set to **ENABLED** (if configured as such) even though the default value is **DISABLED**.

## Delete

Remove a dial-circuit, PPP-FR pseudo device or null device from the list of devices stored in the configuration or to remove user login information. To use the **delete** command, you must have *Administrative* permission.

**Syntax:** `delete interface . . .`  
`user . . .`

### interface *intfc#*

To delete a device, enter the interface or net number as part of the command. To obtain the interface number that the router assigns, use the **list device** command.

If the interface you specify is the last interface in the device list, the **delete** command automatically removes it. If it is not the last interface in the device list, **delete** prompts you to confirm the deletion. The next interface in the list inherits the number of the interface that you deleted, and the remaining interfaces are renumbered (moved up by one position).

If you delete an interface from the middle of the device list, you must enter the CONFIG environments of each of the packet forwarders affected by the deletion. Then, you must modify their configurations to be consistent with the reassigned interface numbers. All device and protocol configuration information related to network interfaces is stored by interface number. Any changes made to interface numbers invalidate some of the device configuration information in the protocols.

Example: `delete interface 2`

```
Are you sure you want to delete this interface?(Y / N)_
```

#### **user *user\_name***

Removes user access to the router for the specified user.

Example: `delete user mary`

```
Delete 'mary'? (Yes or [No]:
```

## **Disable**

Prevents prompting for login data from a remote console and disables modem control. The **disable** command also disables the specified interface.

**Syntax:**    disable    console-login  
                  interface . . .  
                  modem-control

#### **console-login**

Prevents prompting for a user ID and password on the physical console. The default is disable.

Example: `disable console-login`

#### **interface *interface#***

Disables the specified interface after issuing the **restart** command. The default is enabled.

Example: `disable interface 2`

#### **modem-control**

Disables monitoring of modem control lines on the console port. The default is disable.

Example: `disable modem-control`

## Enable

Allow logins from a remote console and enables modem control. The **enable** command also enables the specified interface.

**Syntax:**    enable    console-login  
                  interface . . .  
                  modem-control  
                  ring-monitor-console

### console-login

To provide security to the console, **console-login** enables prompting for a user ID and password on the physical console. If you do not configure an administrative user, and you enable this feature, the following message appears:

```
Warning: Console login is disabled until an
administrative user is added.
```

By disabling the console login, a lock-out situation is prevented.

Example: `enable console-login`

### interface *interface#*

Enables the interface after issuing the **restart** command.

Example: `enable interface 2`

### modem-control

Sets up the router for login on the physical console, if the physical console is connected to the router through a modem. Before using this command, make sure to

- Set your modem for auto-answer.
- Verify that the console baud rate is equal to the modem baud rate.
- Verify that the cable connecting the modem to the router is configured correctly.

The router automatically disconnects the modem when you log out. If your modem disconnects before you log out, the router logs you out.

Example: `enable modem-control`

## Event

Enter the ELS (Event Logging System) environment to display messages on the console. Refer to Chapter 5 for information about the Event Logging System (ELS).

**Syntax:** `event`

Example: `event`

```
ELS config>
```

## Feature

Access configuration commands for specific bridging router features outside the protocol and network interface configuration processes.

Enter a question mark after the **feature** command to obtain a listing of the features available for your software release.

Example: `feature ?`

```
BRS
```

```
Feature name or number [1] ?
```

To access that feature's configuration prompt, enter the **feature** command followed by the feature number or *short name*. Table 2-2 lists available feature numbers and names.

**Table 2–2 Bridging Router Feature Numbers and Names**

<b>Feature Number</b>	<b>Feature Short Name</b>	<b>Accesses the following feature configuration process</b>
0	WRS	WAN Restoral
1	BRS	Bandwidth reservation
2	MCF	MAC Filtering
3	BSA	BOSS SNMP Agent

Example: **feature brs**

```
Bandwidth reservation user configuration
Reservation>
```

After you access the feature’s configuration prompt, you may enter specific configuration commands for that feature. To return to the CONFIG prompt, enter the **exit** command at the feature’s configuration prompt.

**Syntax:** `feature feature# OR feature-short-name`

```
Example: feature 1
         feature BRS
```

## List

Display configuration information for all network interfaces, or configuration information for the router.

**Syntax:** `list` `devices`  
`configuration`  
`patches`  
`obm ip-address`  
`users`  
`v25-bis-address`

### devices

Displays the relationship between a device number and the hardware interface. You can also use this command to check that a device was added correctly using the **add** command.

**Example: list devices**

```
Ifc 0 (FDDI):      CSR      0, vector 0
Ifc 1 (Ethernet): CSR      0, vector 0
Ifc 2 (Ethernet): CSR      1, vector 0
Ifc 3 (Ethernet): CSR      2, vector 0
Ifc 4 (Ethernet): CSR      3, vector 0
Ifc 5 (Ethernet): CSR      4, vector 0
Ifc 6 (Ethernet): CSR      5, vector 0
```

**configuration**

Displays configuration information about the router.

**Example: list configuration**

```
Hostname: lennon 1.98
Maximum packet size: [autoconfigured]
Maximum number of global buffers: [autoconfigured]
Number of Restarts before a Reload/Dump: 64
Logging disposition: detached
Console inactivity timer (minutes): 0
Physical console login: disabled
Modem control: disabled
Contact person for this node: john
Location of this node: Atlantis
Hub-type set to three-slot

Configurable Protocols:
Num Name Protocol
0 IP DOD-IP
3 ARP Address Resolution
4 DN DNA Phase IV
5 XNS Xerox Network Systems*
6 VIN Banyan Vines*
7 IPX NetWare IPX
8 OSI ISO CLNP/ESIS/ISIS
9 DVM Distance Vector Multicast Routing Protocol
10 BGP Border Gateway Protocol
11 SNMP Simple Network Management Protocol
12 OSPF Open SPF-Based Routing Protocol
14 APL AppleTalk
15 DDS Apollo Domain*
22 AP2 AppleTalk Phase 2
23 ASRT Adaptive Source Routing Transparent Enhanced Bridge
24 HST TCP/IP Host Services

Configurable Features:
Num Name Feature
1 BRS Bandwidth Reservation

127296 bytes of configuration memory free
```

\* Not supported in this release.

## patches

Displays the patches and their variables that were implemented using the **patch** command.

Example: **list patches**

Patched variable	Value
ping-size	60
ping-ttl	59
ip-default-ttl	60
ethernet-security	3
rip-static-suppress	3

## obm

Displays out-of-band management port information such as IP address, port speed, and ready-to-send (RTS) enable/disable state (where they apply).

Example: **list obm**

```
OBM Address: 0.0.0.0
OBM Port Speed: 9600
OBM Port RTS disabled
```

**Note:** Setting the **obm-ipaddress** to 0.0.0.0 deletes the current obm address entry.

## users

Displays the users configured to access the system.

Example: **list users**

USER	PERMISSION
joe	operations
mary	administrative
peter	monitor

## v25-bis-address

Displays the v25-bis-address.

Example: **list v25-bis-address**

## Network

Enter the network interface configuration environment for supported networks, such as the FDDI, Ethernet, ISDN, and Token-ring networks.

Enter the interface or network number as part of the command. To obtain the interface number, use the CONFIG **list device** command.

**Syntax:** `network interface#`

Example: `network 2`

```
TKR config>
```

## Patch

Modify the router's global configuration. Patches are recorded in non-volatile configuration memory and take effect immediately. Use this command only to handle uncommon configurations. Use specific configuration commands to modify commonly configured items. The following is a list of the current patches documented and supported for this release.

*ethernet-security*

When set to a non-zero value, zeros the padding that is applied to Ethernet packets whose data portion is less than the physical minimum of 60 bytes. This may be required for security reasons. Default: 0.

*rip-static-suppress*

When set to a non-zero value, static routes are not advertised by RIP out a given interface unless the IP config> **enable send static** command is given for the interface. This changes the semantics of the **enable send static** command. When *rip-static-suppress* is equal to 0 (the default), the list of the routes advertised through RIP is the union of those specified by the interface's RIP flags.

*ping-size*

The size of the data portion (that is, excluding IP and ICMP headers) of the ICMP ping packet that is sent through the IP>**ping** command. Default: 56 bytes.

*ping-ttl*

The TTL (time-to-live) sent in pings by the IP>**ping** command. Default: 60.

*ip-default-ttl*

The TTL used in packets that are originated by the router itself. Default: 60.

**Syntax:** `patch`

Example: `patch`

```
Variable to patch []?  
New value [0]?
```

<i>Variable to patch</i>	The complete name of the patch that you want to change. You cannot use the short syntax.
<i>New value</i>	The new value for the variable that you are patching.

## Protocol

Enter the configuration environment for the protocol software installed in your router. The **protocol** command, followed by the desired protocol number *or* short name, lets you enter a protocol's command environment. After you enter this command, the prompt of the specified protocol appears. From the prompt, you can enter protocol-specific commands. Table 2–3 lists examples of protocol numbers and names. To return to CONFIG, enter the **exit** command.

**Table 2–3 Protocol Numbers and Names**

<b>Protocol Number</b>	<b>Protocol Short Name</b>	<b>Accesses the following protocol process</b>
0	IP	Internet Protocol
3	ARP	Address Resolution Protocol
4	DN	DECnet – a subset of Network Control Program
7	IPX	Novell NetWare Internetwork Packet Exchange
8	OSI	ISO Open Systems Interconnection – Connectionless Network Layer Protocol
9	DVM	Distance Vector Multicast Routing Protocol
11	SNMP	Simple Network Management Protocol
12	OSPF	Open Shortest Path First
14	APL	AppleTalk Phase 1
20	SDLC	SDLC/HDLC Relay
22	AP2	AppleTalk Phase 2
23	ASRT	Adaptive Source Routing Transparent Bridge
24	HST	TCP/IP Host Services
26	DLS	Data Link Switching

**Syntax:** `protocol prot#`

Example: `protocol 7 or protocol ipx`

```
IPX config>
```

## **Qconfig**

Initiate QUICK CONFIG. QUICK CONFIG prompts for information necessary to configure selected devices, bridging protocols, routing protocols, and boot records without entering separate configuration environments.

**Syntax:** `qconfig`

**Note:** For complete information on using the Quick Config software provided with your router platform, refer to the *Quick Configuration* information shipped with your router.

## Set

Configure various system wide parameters.

**Syntax:** `set` `contact-person . . .`  
`data-link . . .`  
`global-buffers`  
`hostname`  
`hub-type. . .`  
`inactivity-timer`  
`location . . .`  
`logging disposition`  
`logging level`  
`obm`  
`packet-size`  
`receive-buffers`  
`restart-count`

### **contact-person** *sysContact*

Sets the name or identification of the contact person for this managed SNMP node. There is a limit of 80 characters for the *sysContact* name length.

Example: `set contact-person nautilus`

### **data-link** *type interface#*

Select the data link type for a serial interface. The *type* can be of the following: PROTEON-SERIAL, PPP, X25, FRAME-RELAY, or SDLC-RELAY. *Interface#* is the number of the interface you are configuring.

Example: `set data-link PPP 3`

### **global-buffers *max#***

Sets the maximum number of global packet buffers, which are the packet buffers used for locally originated packets. The default is to autoconfigure for the maximum number of buffers (up to 1000). To restore the default, set the value to 0. To display the setting for **global-buffers**, use the **list configuration** command.

**Caution:** Use this command only under direct instructions from your Customer Service representative.

Example: `set global-buffers 30`

### **hostname**

Adds or changes the router name. The router name is for identification purposes and it does not affect or change any router addresses.

Example: `set hostname`

### **inactivity-timer**

Changes the setting of the Inactivity Timer. The Inactivity Timer is the period of time that the a remote or physical console may remain inactive before the system logs you out. This command affects only consoles that require login. The default setting of 0 turns the inactivity timer off, performing no logoff, no matter how long a console remains inactive.

Example: `set inactivity-timer`

### **location *sysLocation***

Sets the physical location of an SNMP node. There is a limit of 80 characters for the *sysLocation* name length.

Example: `set location atlantis`

### **logging disposition**

Changes the NVRAM record for the default logging disposition. This command affects the MONITR process (that is, it changes the default setting at startup).

The logging disposition settings, do the following:

- *console* writes to the console (equivalent to the OPCON **divert 2 0** command).
- *detached* holds the data and does not print it (equivalent to the OPCON **halt 2** command).
- *flush* discards the data (equivalent to the OPCON **flush 2** command).

If you have a printing terminal attached to the router's console port, you can obtain a hard copy of the startup messages by setting the logging disposition to **console**, and restarting the router.

Example: `set logging disposition`

#### **logging level**

Controls the output of messages that were not converted to the ELS. The logging level is recorded in the configuration. When you power on or start the router, the logging level takes effect and determines message output. The default logging level is 76.

Example: `set logging level`

#### **obm**

Sets the out-of-band management port ip-address, port speed, and ready-to-send (RTS) enable/disable state (where they apply).

Example: `set obm ip-address`

#### **packet-size**

Establishes or changes the maximum size of a packet buffer.

**Note:** Use this command only under direct instructions from your Customer Service representative. NEVER use it to reduce packet size – ONLY to increase it.

Example: `set packet-size`

#### **receive-buffers max#**

Adjusts the number of private receive buffers for most interfaces. The range is 5 to 255. (On some devices, the maximum value is restricted further, as shown in Table 2–4.) To restore the default, set the value to 0.

The **set receive-buffers** command can be used to increase the receive performance of an interface. In addition, this command can be used to reduce flow control drops when the router forwards many packets from a fast interface to a slow interface. The effect of this command is visible on the **GWCON buffer** command and the **list devices** command.

**Note:** Use this command only under direct instructions from your Customer Service representative.

Example: `set receive-buffers 30`

**Table 2–4 Default and Maximum Settings for Interfaces**

Interface	Default	Maximum
ETH	40	255
Serial	10	40

**Note:** The maximum settings listed in listed in Table 2–4 are hardware dependent. Refer to your hardware installation guide for the values that apply to your specific hardware platform.

#### **restart-count**

Establishes the number of times a router restarts on a serious error before dumping (if enabled) and reloading. In general, do not change the restart-count.

Example: `set restart-count`

## Time

Set the bridging router system time and date and to display the values to the console.

**Syntax:**    time        host ...  
                          list  
                          offset  
                          set ...  
                          sync ...

### host *IP\_address*

Sets the IP address of the RFC868-compliant host that is used as the time source. This is the address of a host that responds to an empty datagram on UDP port 37 with a datagram containing the current time.

Example: **time host 131.210.1.4**  
          IP address of time host [0.0.0.0]?

### list

Displays all configured time-related parameters. This includes the current time (if set) and the source of the time (“operator” or IP address from which time was last received).

Example: **time list**  
          05:20:27 Monday July 26, 1993  
          Set by: operator  
          Time Host: 131.210.4.1                Sync Interval: 10 seconds  
          GMT Offset: -300 minutes

### offset *minutes*

Defines the time zone, minutes offset from GMT (Greenwich Mean Time). Note that values west of GMT are negative. For example, EST is 5 hours earlier than GMT, so the command is “time offset -300.”

Example: **time offset**  
          minutes from GMT (-720 to 720) [0]? -300

**set** <*year month date hour minute second*>

Prompts you to set the current time. If the entire time is not specified in the command, you are prompted for the remaining values.

Example: **time set**

```
year [0]? 1993
month [0]? 7
date [0]? 26
hour [0]? 10
minute [0]? 18
second [0]? 00
```

**sync** *seconds*

Indicates the number of seconds that the router polls the time host for the current time. If you do not set a value for synch, it defaults to 0. Synch must have a value other than 0 for updates to occur.

Example: **time sync 10**

```
seconds between time syncs [0]? 10
```

## Unpatch

Restores the values of the patch variables to their default values.

**Syntax:** unpatch *variable name*

Example: **unpatch ethernet-security**

## Update

Updates the configuration memory when you receive a new software release. Follow the instructions on the release notice sent with the software. The **update** command is the last command that you enter when loading new software. After you enter this command, the console displays a message indicating configuration memory is being updated.

**Syntax:** update *version-of-SRAM*

Example: **update version-of-SRAM**

## CONFIG-ONLY Mode

CONFIG-ONLY mode is used for making configuration changes without running normal operational router software (bridging and forwarding disabled).

CONFIG-ONLY mode uses commands identical to the CONFIG process with the addition of the **restart** and **reload** commands.

The main difference between operational mode and CONFIG-ONLY mode is that operational mode used configuration records to configure the router while CONFIG-ONLY mode assumes the configuration records are corrupt and does not use them.

You enter CONFIG-ONLY mode in the following way:

**Operational Failure.** The router encounters a problem during operation and automatically comes up in CONFIG-ONLY mode. Any of the situations listed below cause the router to enter CONFIG-ONLY mode (see Figure 2–2).

- Repetitive crash detection wherein the router cannot run for more than 5 minutes without crashing. After sixteen such crashes the router attempts to download new router software.
- An unsupported device is in the software load.
- Certain configuration errors have occurred during start up including
  - Non-volatile RAM is corrupted.
  - All router interface information was deleted.
  - Incorrect configuration information was entered.
  - No protocol is configured.



---

## Booting, Dumping, and Upgrading the System

This chapter describes the Boot CONFIG utility. It explains how to use Boot CONFIG, Boot CONFIG Commands, BOOTP Forwarding, and Trivial File Transfer Protocol (TFTP). The information in this chapter is used to boot the router, configure the router for a software upgrade, or to save your configuration database to a remote system.

### Using Boot CONFIG

Router nonvolatile memory includes the data that controls the router boot and dump capabilities. This data is called the configuration database. The Boot CONFIG commands allow you to modify this data.

Using Boot CONFIG commands, you can do the following:

- Add, modify, or remove entries from the boot and dump configuration database.
- Disable or enable network memory dumping and assign a unique name to the dump files.
- Use the TFTP protocol to transfer configuration information between router nonvolatile memory and remote hosts.
- View the current boot and dump configuration database.

- Retrieve the load image from a remote source and load it into the router.

Changes made to system and protocol parameters through Boot CONFIG take effect when you restart the router or when you reload the router software.

## Boot Files

Boot files are the same as load image files. A boot file contains the software load for the router and resides on a host server. The host server can be any device – PC, router, or workstation – that is running the IP protocol and TFTP. The router only knows about the load files that are manually entered into its configuration database. When you enter the **reload** command (at the OPCON prompt), the router loads the image described by the last boot-entry record you entered in the configuration database. Each entry contains the location of the host server and the path, timeout, and filenames for the boot files.

**Note:** In Version 1.1 of the Bridge Router Software, only the most recent (last entered) boot file is used with the **reload** command.

You can use the **add** command to configure more than one boot file in the boot configuration database by specifying the path and name of each boot file. If you have more than one host server, you can use a different host server to boot the router when another host server cannot be reached over the network.

## Entering and Exiting Boot CONFIG

To enter the Boot CONFIG command environment, use the CONFIG **boot** command. This chapter explains how to use this command from the CONFIG process. After you access the Boot CONFIG environment, enter the Boot CONFIG commands at the `boot config>` prompt.

To return to the CONFIG prompt, use the **exit** command.

## Dumping the System Memory Contents

The Bridge Router Software that is loaded on your module includes the capability of dumping the system memory contents for analysis. There are three parts to dump functionality, each of which is controlled by three distinct commands:

- **GWCON dump information** option – Prints information describing the last dump.
- **Boot CONFIG dump-entries**– A database of locations on the network for dumping (to file).
- **OPCON dump** command – Produces a dump of the routers memory contents. This command allows you to test dump-entry records to ensure that the router dumps correctly in the event of a crash.

### Triggering a Dump

The **OPCON dump** command dumps the contents of memory. This command causes the contents of router memory to be dumped to a remote host as specified by one more more existing dump-entry records. Use this command to test your dump-entries to ensure that they work in the event of a system crash.

You manipulate dump-entry records from the OPCON process by entering the following commands:

Command	Result
t 6	sets you into the CONFIG process
boot	sets you at the BOOT CONFIG menu
list dump	prints out all dump entry records
add dump	allows you to add a new dump-entry entry
change dump	allows you to modify an existing dump-entry entry
delete dump	will allow you to remove an existing dump entry entry

The **dump** command verifies that at least one dump-entry record is defined before allowing you to activate the dump procedure.

### Configuring Dumping

An important feature of the Bridging Router is its ability to dump the contents of system memory and the processor's registers to another host during a system reset resulting from a software crash, hardware failure, or user action (pressing the reset switch).

To configure dumping, do the following:

- **Add a dump entry** – This is the location of the host or server that is going to receive the dump file. You add a dump entry with the **add dump-entry command** of the CONFIG process.

The dump code attempts to dump to the first defined dump-entry in the same subnet as the router. If that fails, the dump code then attempts to dump to the next defined record in the same subnet, and so on, until all existing dump-entry records have been attempted. You can view the status of these attempts through the GWCON **dump information** command (not to be confused with the OPCON **dump** (contents of memory) command. The size of the dump file depends upon the amount of memory on a particular platform.

- **Enable dumping** – Dumping does not work unless you enable it using the **enable dumping** command of the CONFIG process. Dumping remains enabled until you use the **disable dumping** command to terminate it.

## Dump Files

When the router crashes and dumping is enabled, the router writes the memory contents to a remote host using TFTP. Each dump entry contains the location of the host server and the path, timeout, and filenames for the dump files.

You can configure the router to automatically append a unique character string to the dump filenames to prevent subsequent dumps from overwriting an existing dump file. See the **enable unique-naming** and **disable unique-naming** commands in this chapter.

Dump files are for diagnostic purposes only. Enable the router's dump and unique-naming capabilities only on the advice of your Customer Service representative.

## Boot and Dump Directories

You must create directories to contain the boot and dump files. These directories must reside on a host server. The boot directories must be globally readable and the dump directories globally writeable. The boot and dump functions use the TFTP protocol. Your TFTP server may impose additional restrictions.

## Displaying Dump Information

You obtain dump information from the GWCON (+) prompt, using the DUMP command. Refer to Chapter 1 for information about entering the GWCON process.

The Dump command prints out a list of eight possible dump entries and information detailing the status of the last dump performed by the system. This status information appears as in the following example:

```
1: Dump Failed to "/usr/router1.dmp" on 1.2.3.4
   error = TFTP protocol error
2: Dump Not Attempted to "/usr/tftp/router1.dmp" on 10.23.2.5
3: Dump Successful to "/usr/tftp/router1.dmp" on 1.2.3.4
```

Possible outcomes for each dump entry are:

- Not Attempted
- Failed
- Successful

**Note:** By design, in the example above all configured dump entries are displayed, not just those up to and including the successful dump location.

If the IP address of a gateway is different from the default (0.0.0.0), then that information is printed, otherwise it is left blank.

If a dump failed (status = Failed), an appropriate error message ("string") is displayed. Possible error messages are listed below:

- Invalid ROMBLKOUT
- Invalid request passed via ROMBLKOUT
- No dump requested
- No interface with IP protocol
- Unattended dump is disabled by config recs
- All dump records have been tried

- Dump not supported on interface type
- Interface board is not present
- Network device cannot be initialized
- TFTP Timeout
- TFTP protocol error

## Boot CONFIG Commands

This section describes the Boot CONFIG commands. Each command includes a description, syntax requirements, and an example. Table 3–1 summarizes the Boot CONFIG commands.

**Note:** Not all Boot CONFIG commands apply to all hardware platforms.

**Table 3–1 Boot CONFIG Commands**

Command	Function
<b>? (Help)</b>	Displays a list of the commands available from that prompt level.
<b>Add</b>	Adds a network boot interface, an IP address to a specified interface, host boot entry, host dump entry, or bootp device.
<b>Change</b>	Changes the boot interface IP address, network boot entry data, network dump entry data, or bootp device.
<b>Delete</b>	Deletes a network boot interface, host boot entry, host dump entry, address, or bootp device.
<b>Disable</b>	Disables memory dump or unique naming of the dump files.
<b>Enable</b>	Enables memory dump and allows unique naming of dump files.
<b>List</b>	Displays all network boot addresses, all boot and dump configuration data, and name settings.
<b>Load</b>	Copies the boot file to RAM, copies the boot file from a remote host, or copies a new software image from a remote host.
<b>TFTP</b>	Initiates TFTP file transfers between router memory and remote hosts.
<b>Exit</b>	Leaves the Boot CONFIG environment and returns to the CONFIG process.

## ? (Help)

Lists the commands that are available from the current prompt level. You can also enter a ? after a specific command name to list its options.

**Syntax:** ?

Example: ?

```
ADD
CHANGE
DELETE
DISABLE
ENABLE
LIST
LOAD
TFTP
EXIT
```

Example: **add** ?

```
ADDRESS
BOOT-ENTRY
BP-DEVICE
DUMP-ENTRY
```

## Add

Creates an entry in the boot and dump configuration database.

**Syntax:** add address  
boot-entry  
bp-device  
dump-entry

### address

The IP address of the interface or device over which the router can boot or dump. When you enter the **add address** command, you must supply the value for the following information (or accept the default):

- Interface number of the network interface
- IP address

- Network mask

**Note:** To obtain the interface number (Ifc#), use the CONFIG **list devices** command. Chapter 3 describes this command. You must provide an address in order for the router to boot or dump over the network.

Example: **add address**

```
Which interface is this address for [0]?
New address [0.0.0.0] ?
Net mask for this interface [255.255.255.0]?
```

#### boot-entry

Enables you to enter information related to the boot server's location and boot file name. You can configure more than one boot device. When you enter the **add boot-entry** command, you must supply the following information:

- IP address of the remote host.
- IP address of the first hop router, if any.
- TFTP retransmission timer value. The default is 10 seconds. You may need to set this value to a longer time over exceptionally slow boot paths.
- Path and filename.

**Note:** To view a list of the current boot configuration, enter the Boot CONFIG **list boot** command.

Example: **add boot-entry**

```
remote host [0.0.0.0]? 10.0.0.5
through gateway (0.0.0.0 if none) [0.0.0.0]? 10.0.0.7
timeout in seconds [10] 10?
file name [ ]? /loads/routeabout.ldc
```

#### bp-device

Adds the specified interface as a BOOTP (Boot Protocol) device. The BOOTP device is the interface that BOOTP requests are sent through if the router image becomes corrupted. A BOOTP client sends out its hardware address, requesting its IP address, the IP address of a host that is willing to load it, and the load filename. There must be a BOOTP server set up on the LAN (or reachable from that LAN) designated as the BOOTP device for this to work. Once this information is obtained, the client uses TFTP to obtain the load file.

Example: **add bp-device**

```
Which interface number [0]?
```

### **dump-entry**

The IP address of the remote host that receives the dump files. When you enter the **add dump-entry** command, you must supply the following information:

- IP address of the remote host.
- IP address of the first hop router (if any).
- TFTP retransmission timer value.
- Base dump filename (may have unique suffix appended).

**Note:** To view a list of the dump configurations, enter the Boot CONFIG **list dump** command. The **GWCON dump** command lists the status of the last dump attempted from this router. It uses data from the dump-entry records to display this information. However, if changes are made to any dump-entry records, the displayed information is invalid.

Example: **add dump-entry**

```
remote host [0.0.0.0]? 1.2.3.4
through gateway (0.0.0.0 if none) [0.0.0.0]?
timeout in seconds [10]?
file name []? /usr/tftp/router11.dmp
```

### **Change**

Modify an entry in the boot and dump configuration database. Changes an existing address for a boot interface or device that was previously added.

**Syntax:**    change    address  
                  boot-entry  
                  bp-device  
                  dump-entry

### **address**

Prompts you to provide new information to modify an existing address for a boot interface or device. When you respond to the prompts from the **change address** command, you must supply the following information:

- Address entry number
- Interface number of the network interface
- IP address
- Net mask

**Note:** When you enter the Boot CONFIG **list** command, the console displays some of this information, such as the address entry number. To obtain the interface number (Ifc#), use the CONFIG **list devices** command. (The previous chapter describes this command.)

Example: **change address**

```
Change which entry [1]?
Which interface is this address for [0]?
New address [192.9.1.1]?
Net mask for this interface [255.255.255.0]?
```

### **boot-entry**

Prompts you to provide new configuration information for a previously added network boot file. When you enter the **change boot-entry** command, you must supply the following information:

- Boot entry number
- IP address of the remote host
- IP address of the first hop router, if any
- TFTP timeout value, if any
- Boot file name, if different from the current file name

**Note:** When you enter the Boot CONFIG **list boot-entries** command, the console displays some of this information, such as the boot entry number.

Example: **change boot-entry**

```
change which entry [1]?
remote host [18.123.0.16]?
through gateway (0.0.0.0 if none) [0.0.0.0]?
timeout in seconds [3]?
file name [user/lib/gw/gwimage.ldb]? /user/lib/gw/gwimag1.ldb
```

### **bp-device**

Prompts you to provide a new number for the interface that is the BootP device. To obtain the entry number for an interface, use the **list boot-entries** command.

Example: **change bp-device**

```
Change which entry [1]?
Which interface is this entry for [1]?
```

**Note:** For more information on the BootP protocol and its related functions, refer to the section, “The BootP Forwarding Process,” in this chapter.

### **dump-entry**

Prompts you to provide new information for a previously added network dump file. When you enter the **change dump-entry** command, you must supply the following information:

- Dump entry number
- IP address of the remote host
- IP address of the first hop router, if any
- TFTP retransmission timer value
- Dump file name, if different from the current file name

**Note:** Use the Boot CONFIG **list dump-entries** command to display this information.

Example: **change dump-entry**

```
change which entry [1]?
remote host [18.123.0.16]?
through gateway (0.0.0.0 if none) [0.0.0.0]?
timeout in seconds [3]?
file name [user/lib/gw/gwimage.ldb]? /usr/tftp/router11.dmp
```

## Delete

Remove entries from the boot and dump configuration database.

**Syntax:**    delete     address  
                  boot-entry  
                  bp-device  
                  dump-entry

### address

Prompts you to enter the number of the interface address of the entry you want to delete from the boot and dump configuration database.

Enter the number of the interface address that appears under the *Interface addresses* heading when you enter the Boot CONFIG **list address** command.

Example: **delete address**

```
Delete which entry [1]?
```

To verify the deletion, use the **list** command.

### boot-entry

Prompts you to enter the number of the boot entry you want to delete from the boot and dump configuration database. The boot entry number appears under the *Boot files* heading when you enter the Boot CONFIG **list boot-entries** command.

Example: **delete boot-entry**

```
Delete which entry [1]?
```

To verify the deletion, use the **list** command.

## bp-device

Prompts you to enter the number of the interface of the BootP device you want to delete.

Example: `delete bp-device`

```
Delete which entry [1]?
```

**Note:** For more information on the BootP protocol and its related processes, refer to the section “The BootP Forwarding Process” in this chapter.

## dump-entry

Prompts you to enter the number of the dump entry you want to delete from the boot and dump configuration database. The dump entry number appears under the *Dump to* heading when you enter the Boot CONFIG **list dump-entries** command.

Example: `delete dump-entry`

```
Delete which entry [1]?
```

To verify the deletion, use the **list** command.

## Disable

Disables network memory dumping and the unique naming of dump files.

**Syntax:** `disable` `dumping`  
`unique-naming`

## dumping

Suppresses the network memory dump feature of the router when a system failure occurs. After you enter this command, use the Boot CONFIG **list all** command to verify the dump settings. Dumping is suppressed until you enter the **enable dumping** command (the default setting).

**Note:** This function is used for unattended dumping only. The OPCON **dump** command ignores this setting.

Example: `disable dumping`



## Exit

Leave the current process command level and return to the previous process level.

**Syntax:** `exit`

Example: `exit`

## List

Displays the current boot and dump configuration database.

**Syntax:** `list` addresses  
all  
boot-entries  
bp-device  
dump-entries

### addresses

Displays the IP addresses and their subnet masks of all of the network boot interfaces.

Example: `list addresses`

```
Interface addresses:  
1: 192.9.1.1 on interface 0, mask 255.255.255.252  
2: 192.9.223.39 on interface 5, mask 255.255.255.0
```

### all

Displays all boot and dump configuration data and the current settings for the dump and unique-naming capabilities.

Example: **list all**

```
Interface Addresses:
 1: 128.185.208.3 on interface 0, mask 255.255.255.0
Boot files:
 1: "gw/routeabout.13.0Z.ldc" on 128.185.123.17 through 128.185.208.49
for 3 secs
BOOTP over interface(s): 3
Dumping disabled
Unique-naming disabled
Dump to:
 1: "/usr/local/tftp/router1.dmp" on 1.2.3.4 via 0.0.0.0 for 10 secs
 2: "/usr/tftp/router1.dmp" on 13.12.2.3 via 13.12.245.7 for 10 secs
```

### **boot-entries**

Displays the boot file configuration.

Example: **list boot-entries**

```
1: /usr/lib/gw/this-dn.ldb on 192.9.1.2 through 0.0.0.0 for 3 secs
2: /usr/lib/gw/this.ldb on 192.9.2.2 through 192.9.1.4 for 3 secs
```

### **bp-device**

Lists the interfaces that were previously added as BOOTP devices.

Example: **list bp-device**

```
BOOTP over interface(s): 0 1
```

### **dump-entries**

Displays the dump file configuration.

Example: **list dump-entries**

```
1: "/usr/local/tftp/router1.dmp" on 1.2.3.4 via 0.0.0.0 for 10 secs
2: "/usr/tftp/router1.dmp" on 13.12.2.3 via 13.12.245.7 for 10 secs
```

## **Load**

**Syntax:** load remote

### **remote**

Interactively prompts for boot information and attempts to obtain the specified file using TFTP. If the download is successful, the file is loaded into FLASH. To perform a remote load do the following:

1. Enter the **load remote** command after the `Boot config>` prompt and enter the remote host address, remote pathname, first hop address, and TFTP timeout value after the prompts.
2. When the prompt asks you to confirm the load, enter **no** to cancel the command, or enter **yes** to load the boot file.

Example: **load remote**

```
Remote Host Address[0.0.0.0]? 1.2.3.4
Remote Pathname[ ]? /usr/tftp/router11.ldc
First Hop Address[0.0.0.0]?
TFTP Timeout Value[3]?
```

3. After you enter the information above, the **load remote** command prompts you to confirm the action. Choose **Yes** to cause the router to restart in MOS and attempt to load the image (See the “Starting the Upgrade” section in this Chapter). Choose **No** to cancel the operation.

```
Are you sure you want to reload the gateway(Yes or No): yes
```

<i>Remote Host Address</i>	IP address of the host containing the boot file.
<i>Remote Pathname</i>	Pathname and filename of the boot file you want to load.
<i>First Hop Address</i>	The address of the first-hop router that routes to other networks. This is needed if the remote host address is not on a directly-connected network
<i>TFTP Timeout Value</i>	Specifies the time interval between the TFTP packet retransmissions. Longer values (longer than the default value of 3) may be needed when booting over or across slow networks or serial lines.

## Trivial File Transfer Protocol (TFTP)

TFTP is a file transfer protocol that runs over the Internet UDP protocol. This implementation provides multiple, simultaneous TFTP file transfers between a router’s non-volatile configuration (NVRAM) and remote hosts.

TFTP transfers involve a *client* node and a *server* node. The client node generates a TFTP request on the network. The router acts as a client node by generating TFTP requests from the router console using the Boot CONFIG process **TFTP** command.

The client can transfer a copy of the configuration file stored in non-volatile config memory from a TFTP server.

The server is any device (for example, PC, router, workstation) that receives and services the TFTP requests. When the router acts as a server, transfers are transparent to the user. Use the ELS message log to view the transfer in progress.

**Note:** For security reasons, this implementation of TFTP does not allow any remotely initiated writes (put) to the router.

Before using the **TFTP** command, note the following:

- The router configuration must include the IP protocol and have at least one configured IP address. Also, the router must not be operating in config-only mode.
- When a router's NVRAM memory is empty (that is, initially installing the router, or if corrupted NVRAM is deleted), you must set the following parameters in order to restore the router's configuration.
  - Set the router's hostname.
  - Configure IP so that the router can reach each host with the archived configuration. The *Routing Protocols User's Guide* explains the IP configuration commands.
- The source IP address for TFTP transfers is the router ID. By default, this ID is a configured IP address for one of the router's network interface cards. To change the router ID, use the **set router ID** command at the `IP Config>` prompt.
- All TFTP data transfers are 512 bytes long. A data transfer of less than 512 bytes indicates an end to the transfer. A protocol, client, or remote host error generates an error packet that terminates the transfer.
- You must download configuration files into the same type of router from which you are uploading the file.

Every TFTP transfer has a client and server UDP port number. When a client node generates an initial request to the server, an unused UDP port number on the client node is randomly selected as the client port. The server port is the UDP port number 69 (decimal). If a TFTP server is running on the server, it listens on UDP port 69. When the server receives a request from the network, a UDP port number currently unused on the server is randomly selected as the host port. The file transfers then occur on these two UDP ports.

## TFTP

Initiates TFTP file transfers between a remote host and the router's non-volatile configuration (CONFIG). TFTP is also used to download new system software for the router.

**Note:** "CONFIG" is a special file name assigned for storage of non-volatile configuration memory in the router.

We can use TFTP to save (put) and restore (get) configuration information to and from remote hosts. The router operating system also uses TFTP to reload new software images, and the router DUMP functionality uses a different implementation of TFTP to save the state of the router and the contents of memory for later analysis.

Here, the router acts as a TFTP client. The remote host is any device (for example, including router, workstation, PC) running IP that acts as a TFTP server node.

The **TFTP get** and **put** commands lock the CONFIG process for the duration of the operation. The following two keyboard characters are recognized during the TFTP operation:

- |                     |                                |
|---------------------|--------------------------------|
| <code>CTRL/p</code> | Displays the OPCON prompt (*). |
| <code>CTRL/c</code> | Cancels the TFTP operation.    |

**Note:** Do not press the reset switch or power off the router while it is performing a **TFTP get** operation. This may leave the configuration memory in an inconsistent (and invalid) state.

**Syntax:**    \_tftp            \_get  
                                  \_put

### get

Requests that a TFTP server transfer a file *from* the server *to* the router. The server sends a data packet and the client acknowledges receipt of the data. This cycle continues until the transfer is complete

Example: `tftp get config 11.22.33.44 /usr/local/tftp/router11.cfg`

or

Example: `tftp get`

```
Local filename [CONFIG]? config
Remote host [0.0.0.0]? 11.22.33.44
Host filename []? /usr/local/tftp/router11.cfg
```

<i>Local filename</i>	Currently, the only valid file name is CONFIG.
<i>Remote Host</i>	Specifies the address of the host containing the file you want to transfer.
<i>Host filename</i>	Specifies the name of the file on the host that you want to transfer. Make sure that you specify the <b>complete</b> pathname.

The following message appears on the console when the transfer completes:

```
TFTP transfer complete, Status: OK
```

If the TFTP transfer is unsuccessful, a detailed error message appears on the screen. While transferring a file to NVRAM, the following message appears on the console:

```
Updating CONFIG: Do Not Interrupt!
```

If you attempt to transfer a file, and there is not enough memory to buffer the file, the following message appears on the console:

```
TFTP transfer complete, Status: Out of Memory
```

**Caution:** Do not reset or power off the router while updating of the CONFIG is in progress. This may corrupt the data in NVRAM forcing you to reconfigure the router.

### Magic Numbers

There are three tests for validating an image before the process writes it into the router's NVRAM memory. First, the router assigns a *Magic Number* to each platform type for the archived image and the image to be restored. If the numbers do not match, the process aborts the transfer and the console displays the following message:

```
Bad Magic Number
```

In the second test, the host name for the router that originally archived the image is compared to the host name for the router that is restoring the image. If the host names do not match, you are prompted

```
Is this acceptable? (Yes or [No])
```

If you answer Yes, the transfer proceeds; if you answer No, the transfer is aborted.

The third test checks for adequate available memory. If there is not enough memory to accommodate the transfer and its accompanying operations, the transfer fails. When a transfer fails due to insufficient available memory, the console displays one of the following messages:

```
Out of Memory.
```

or

```
Unable to allocate buffer memory. Config not updated.
```

**Note:** All of the tests listed above must be successful (pass) before the existing configuration is deleted and the new configuration is written to NVRAM.

### put

Requests that a TFTP server transfer a file *to* the server *from* the router. The server acknowledges the request and the client transfers the file. This cycle continues until the transfer is complete.

The console display is the same as for the **TFTP get** command.

Example: `tftp put config 11.22.33.44 /usr/local/tftp/router11.cfg`

or

Example: `tftp put`

```
Local filename [CONFIG]? config
Remote host [0.0.0.0]? 11.22.33.44
Host filename []? /usr/local/tftp/router11.cfg
```

The following message appears on the console when the transfer is complete:

```
TFTP transfer complete, Status: OK
```

The **TFTP** command generates the following error messages:

<b>Error Message</b>	<b>Meaning</b>
Unknown Error	Protocol failure.
File Not Found	Specified host file does not exist.
Access Violation	File protection error.
Disk Full	File system full during write.
Illegal Operation	Undefined TFTP operation requested.
Unknown TID	Unexpected TFTP packet received.
File Already Exists	File already exists.
No Such User	TFTP not supported on host.

## The BOOTP Forwarding Utility

BOOTP (documented in RFC 951) is a bootstrap protocol used by a router or a diskless workstation to learn its IP address, the location of its boot file, and the location of its boot server. The Bridging Router can act as a *BOOTP client* or as a *BOOTP relay agent* for another device. The following sections describe these two processes.

### The Bridging Router as a BOOTP Client

The router acts as a BOOTP Client when it needs to find the location of the boot file and boot server. You can configure the Bridging Router's boot PROM to have the router act as a BOOTP Client. It can also be a BOOTP Client if, during booting, it cannot find the boot file and server. When either of these conditions exist, the router broadcasts a UDP packet over one of its LAN interfaces (it looks for the LAN interface that completes its self-test first, usually ethernet) to the *BOOTP server* that contains the pathname of the boot file and server. The following steps illustrate this process:

1. The BOOTP client copies its MAC address (or ethernet address if it is booting over an Ethernet LAN) into a BOOTP packet (UDP packet) and broadcasts it onto the local LAN. BOOTP is running on top of UDP.
2. The BOOTP server receives the request and looks up the client's Ethernet address in its database. If found, it formats a BOOTP reply containing the client's IP address, subnet mask, and boot server name. The reply is sent back over the LAN to the BOOTP client.

**Note:** If multiple hops are required before reaching the BOOTP server, a BOOTP relay agent receives the packet. BOOTP relay agent is explained in the next section.

3. When the router receives the BOOTP reply packet it uses the information to initiate a TFTP request to the boot server.

### The Bridging Router as a BOOTP Relay Agent

If the BOOTP request needs multiple hops before reaching the BOOTP server, the BOOTP relay agent routes the packet through IP to all BOOTP servers that it knows about. If any other router receives this packet while it is being routed through IP, it examines the packet to determine whether it is a BOOTP packet and routes that packet to the BOOTP servers that it knows about. The following steps describe the BOOTP relay agent forwarding process.

1. Acting as the local BOOTP relay agent, the bridging router receives the BOOTP request packet from the BOOTP client. It then modifies the checksum, places an IP header on the packet, and routes the packet to all BOOTP servers
2. The BOOTP server receives the request and looks up the client's MAC address in its database. If found, it formats a BOOTP reply containing the client's IP address, subnet mask, and boot server name. The reply is then sent to the BOOTP relay agent.
3. The BOOTP relay agent receives the reply and makes an entry in its ARP table for the client and then forwards the reply to the BOOTP client.
4. The client then continues to boot using the information that is contained in the BOOTP reply packet to initiate a TFTP request to the boot server.

## BOOTP Server

The BOOTP server is either a UNIX host with a *bootpd* daemon or a DOS host (running software available from FTP Software). The BOOTP server contains a file (maintained by the network administrator) that lists all the BOOTP clients that this server is responsible for, and their associated IP addresses, and the locations and names of their boot files.

When the BOOTP server receives a BOOTP request it compares the MAC address of the client with the MAC address in its BOOTP file. If a match occurs, the server constructs a BOOTP reply and adds the client's IP address, along with the location of the Boot server and boot filename. If a match does not occur, the packet is dropped.

## Enabling/Disabling BOOTP Forwarding

To enable or disable BOOTP forwarding on the router, enter the following command line at the IP configuration prompt.

```
IP Config> enable bootp  
IP Config> disable bootp
```

When enabling BOOTP, you must enter the following values:

- Maximum number of application hops you want the BOOTP request to span. This is the maximum number of BOOTP relay agents that can forward the

packet. This is NOT the maximum number of IP hops to the BOOTP server. A typical value for this parameter is 1.

- Number of seconds you want the client to retry before you forward the BOOTP request. *This parameter is not commonly used.* A typical value for this parameter is 0.

After accepting a BOOTP request, the router forwards it to each BOOTP server. If there are multiple servers configured for BOOTP, the transmitting server replicates the packet.

## Configuring a BOOTP Server

To add a BOOTP server to the router's configuration, enter the following command at the IP configuration prompt:

```
IP Config> add BOOTP-SERVER [IP address of server]
```

You can configure multiple servers. In addition, if you know only the network number of the server or if multiple servers reside on the same network segment, you can configure a broadcast address for the server.

## Upgrading the System Software

Digital router modules are designed with a facility that enables you to obtain a copy of the operating system software from a remote host. You can use this facility to upgrade the system software to a more recent release or to reload the current release.

When you invoke the upgrade function through the **reload** or **load remote** commands, the resident image restarts in IP Host-only mode and the router attempts to obtain a new load image using TFTP. IP Host-only mode causes the router to restart as an IP end station (no routing or bridging enabled), which allows the router to devote all its resources to loading the new image.

If you invoke the **reload** command, the router attempts to load the file specified by the most recent (highest numbered) boot-entry configuration record. If you invoked the **load remote** command, the router attempts to obtain the image specified by the data entered on the command line.

To ensure that the load host is reachable (because the router is now an IP end station), the router automatically installs a static route to the specified load host in the routing table using the data specified by the load request (either a boot-entry or command line data).

When the TFTP transfer of the new image completes successfully, the received image is validated, then written to FLASH. When the FLASH write is complete (which may take several minutes), the router restarts the new image using the existing configuration information.

If the TFTP transfer fails for any reason, the router restarts the original resident image using the existing configuration information. If the FLASH write fails for any reason (power failure, for example) while overwriting the old image, the router is left without a valid image. In this case, a platform specific PROM-based primitive loader attempts to obtain a new image using BOOTP or TFTP.

## Starting the Upgrade

You activate the upgrade function through the command line interface using the OPCON **reload** command (see Chapter 1) or the Boot CONFIG **load remote** command. When the upgrade procedure initially restarts MOS, it displays the following notice:

```
[IP Host-Only Mode - Upgrading Operational Image]
```

During the upgrade procedure, the following output appears on the monitor:

```
*reload
Are you sure you want to reload the Gateway? (Yes or [No]): y

System Restart ...
.
.
Copyright Notices:
Copyright 1995 Digital Equipment Corp.
Copyright 1985-1994 Proteon, Inc.
Copyright 1984-1987, 1989 by J. Noel Chiappa

MOS Operator Control

[IP Host-Only Mode - Upgrading Operational Image]

Network FLASH update proceeding...
```

After a wait of 30 seconds (to assure that the router initialization is complete), the following message appears:

initialization complete

**Note:** Between the start of the upgrade TFTP transfer and the FLASH update, you can press `CTRL/p` to enter OPCON from which you can enter any other process, for example, to **talk** to process 2 to view the status of the TFTP transaction for diagnostic purposes. Display must be enabled for the TFTP subsystem in the ELS. If display is not enabled you can enter process 5 and enable it.

If you do switch to another process, you cannot view the output of the CONFIG process because there is now no device connected to the CONFIG process. The upgrade is not blocked, however, and completes normally. Unless you need to switch to another process, remain in process 6.

You can still cancel the upgrade operation by pressing `CTRL/c`

### The reload Command

The OPCON **reload** command obtains the upgrade parameters (TFTP remote host IP address, gateway IP address, timeout value in seconds, and remote file name) from a predefined boot-entry configuration record. The **reload** command checks to make sure that at least one boot-entry is defined before it is activated. If no boot-entries are defined, the following message is displayed:

```
Aborted, No boot entries defined Configure a boot entry or use LOAD
REMOTE
```

If the **reload** command finds a boot-entry, it prompts you with the message:

```
Are you sure you want to reload the gateway? (Yes or [No])
```

### The load remote Command

The **load remote** command obtains the necessary TFTP information through a series of prompts issued in response to the command. Refer to the section, “Boot CONFIG Commands,” for information about the **add boot-entry** command and the **load remote** command.

To use the upgrade feature, you must have admin privileges.

### IP Host-only Mode

IP Host-only mode is a restart state that the router enters as a result of repetitive crash detection or as part of the upgrade procedure.

When restarted in IP Host-only mode, the router becomes an IP end station. Only UDP, TCP, ARP, TELNET, TFTP, and SNMP are enabled. IP routing, bridging, RIP, OSPF, ISIS, EGP, BGP, and all other protocols are disabled. During normal operation, most RAM is allocated as buffers for protocol forwarders. During an upgrade, a large amount of RAM is needed to receive the image file being copied onto the router by TFTP. With the forwarders disabled, this RAM is available to receive the image file.

**Note:** In HST mode (bridge only, no IP routing), ASRT bridging is enabled so that IP Host Services are available.

## Crash Handler

Repetitive crash detection defines three distinct severities: normal, excessive, and extreme. Each state is defined by a maximum number of crashes within a fixed time interval. If the number of crashes in the time interval exceeds the number allowable for the severity, the severity is escalated to the next higher level.

The crash severity in effect at the time of a crash (at the time of a reboot following a crash) determines how the system is rebooted. The severities and their corresponding reboot personalities are:

- **Normal** – Normal MOS startup (full configured protocol init).
- **Excessive** – IP host-only mode MOS startup (minimal configured protocol initialization).
- **Extreme** – CONFIG-ONLY mode MOS startup (no configured protocol initialization).

When the crash detection handler reports unrecoverable image problems, one further severity condition exists, the Image Corrupt severity level:

- **Image Corrupt** – Operational image checksum intentionally corrupted and control passed to boot PROM. This results in a platform BOOTP load request sent on the default interfaces. It is an ultimate attempt to reload a new operational image and requires a suitably configured BOOTP network server and load file.

---

## Monitoring the System

This chapter describes the GWCON process. It explains how to use GWCON and discusses its commands.

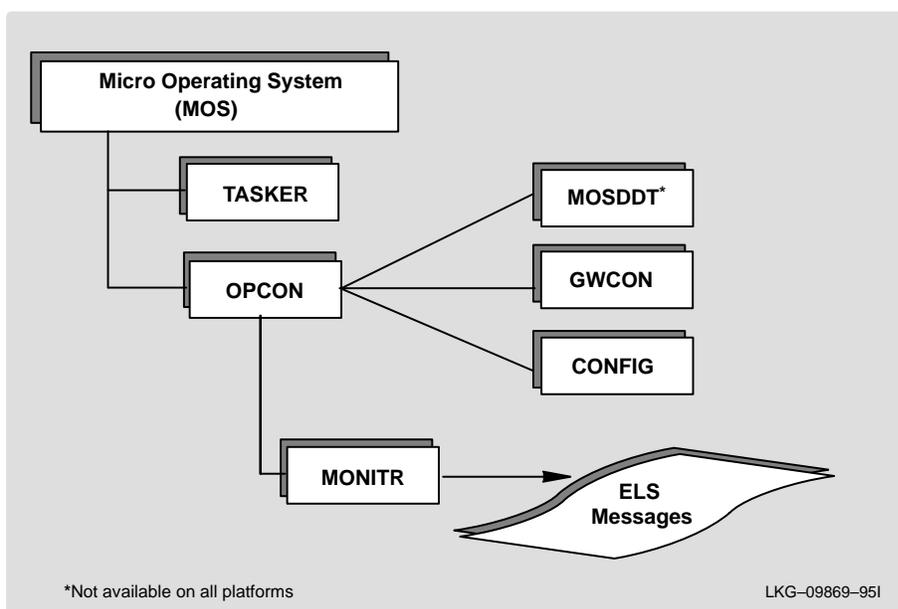
### Using GWCON

The Gateway Console process, GWCON (also referred to as CGWCON), is a second-level process of the router user interface. Using GWCON commands, you can do the following:

- List the protocols and interfaces currently configured in the router.
- Display memory and network statistics.
- Set current ELS parameters.
- Test a specified network interface.
- Configure volatile configuration parameters.
- Enable and disable interfaces.

GWCON fits into the router software structure as shown in Figure 4–1.

Figure 4–1 Config in the Router Software Structure



The GWCON command interface consists of a group of commands. Some of these commands, such as **clear** perform actions. Some commands, such as **network**, invoke a command interpreter that allows you to execute a subset of commands. Each group of commands has its own prompt.

If you want to know the process you are communicating with, press **RET** to display the prompt. CONFIG prompts usually appear as `xxx Config>`, where `xxx` is the protocol you are configuring; GWCON prompts appear as `xxx>`.

## Entering and Exiting GWCON

To enter the GWCON command environment from OPCON and obtain the GWCON prompt do the following:

1. At the OPCON prompt, enter the **status** command to find the pid of GWCON.

\*s

Pid	Name	Status	TTY	Comments
1	COpCon	IOW	TTY0	
2	Monitr	DET	--	
3	Tasker	IDL	--	
4	MOSDDT	DET	--	
5	CGWCon	IOW	--	
6	Config	IOW	--	
7	ROpCon	IOW	TTY1	128.183.142.17
8	ROpCon	RDY	TTY2	128.182.178.5

\*

2. Enter the **talk** command and the pid for GWCON to get to the GWCON process.

```
*talk 5
```

The console displays the GWCON prompt (+). Now you can enter GWCON commands.

If the prompt does not appear, press **RET**.

To return to OPCON, enter the OPCON intercept character (the default is **CTRL/p**).

## GWCON Commands

This section contains the GWCON commands. Each command includes a description, syntax requirements, and an example. After accessing the GWCON environment, enter the GWCON commands at the + prompt. Table 4-1 summarizes the GWCON commands.

**Table 4–1 GWCON Command Summary**

<b>Command</b>	<b>Function</b>
<b>? (Help)</b>	Lists the GWCON commands.
<b>Boot</b>	Displays boot information for the bridging router.
<b>Buffer</b>	Displays information about packet buffers assigned to each interface.
<b>Clear</b>	Clears statistics.
<b>Configuration</b>	Lists the current protocols and interfaces.
<b>Disable</b>	Takes the specified interface off line.
<b>Dump</b>	Displays information about the last system dump.
<b>Error</b>	Displays error counts.
<b>Event</b>	Enters the Event Logging System environment.
<b>Feature</b>	Provides access to console commands for independent router features outside the usual protocol and network interface console processes.
<b>Interface</b>	Displays network hardware statistics or statistics for the specified interface.
<b>Log</b>	Sets or views the logging level for events not included in the Event Logging System.
<b>Memory</b>	Displays memory, buffer, and packet data.
<b>Network</b>	Enters the command environment of the specified network.
<b>Protocol</b>	Enters the command environment of the specified protocol.
<b>Queue</b>	Displays buffer statistics for a specified interface.
<b>Statistics</b>	Displays statistics for the specified interface.
<b>Test</b>	Enables a disabled interface or tests the specified interface.
<b>Uptime</b>	Displays time statistics for the router.

## ? (Help)

Lists the commands that are available from the current prompt level. You can also enter a ? after a specific command name to list its options.

**Syntax:** ?

Example: ?

```
BOOT information
BUFFER statistics
CLEAR statistics
CONFIGURATION of gateway
DISABLE interface
DUMP information
ERROR counts
EVENT logging
FEATURE commands
INTERFACE statistics
LOG level
MEMORY statistics
NETWORK commands
PROTOCOL commands
QUEUE lengths
STATISTICS of network
TEST network
UPTIME of gateway
```

Example: **protocol** ?

```
IP
ARP
DN
IPX
DVMRP
DLSW
OSI
SNMP
OSPF
APL
SDLC
APL2
SRT
HST
```

## Boot

Displays boot information for this bridging router.

**Syntax:** boot

Example: **boot**

```
Booted using Ethernet, line 0 at (80740000, 4) as 128.185.227.220
Filename gw/mwWD.ldc
Host 128.185.122.17, Gateway 128.185.227.15
```

**Note:** The actual display may vary depending on your hardware platform.

**Note:** Refer to your hardware installation guide for information about the boot process for your specific hardware module.

## Buffer

Displays information about packet buffers assigned to each interface.

**Note:** Each buffer on a device is the same size and is built dynamically. Buffers vary in size from one device to another.

To display information about one interface only, enter the interface or network number as part of the command. To obtain the interface number, use the **GWCON configuration** command.

**Syntax:** buffer *network#*

Example: **buffer 1**

Nt	Interface	Input Buffers				Buffer sizes					Bytes	
		Req	Alloc	Low	Curr	Hdr	Wrap	Data	Trail	Total	Alloc	
0	TKR/0	20	20	7	0	109	92	2052	7	2260	45200	
1	TKR/1	20	20	7	20	109	92	2052	7	2260	45200	
2	FDDI/0	40	40	10	40	111	92	4478	19	4700	188000	
3	SL/0	10	10	4	0	108	92	2048	0	2248	22480	
4	Eth/0	40	40	10	40	108	92	1500	4	1704	68160	
5	Eth/1	40	40	10	40	108	92	1500	4	1704	68160	

*Nt* Network interface number associated with the software.

*Interface* Type of interface.

*Input*

*Buffers:* Number of buffers requested.

*Req*

<i>Alloc</i>	Number of buffers allocated.
<i>Low</i>	Low water mark (for flow control).
<i>Curr</i>	Current number of buffers on this device. The value is 0 if the device is disabled. When a packet is received, if the value of <i>Curr</i> is below <i>Low</i> , then the packet is eligible for flow control. (See the <b>queue</b> command for conditions.)

*Buffer Sizes:*

<i>Hdr</i>	Sum of the maximum hardware, MAC, and data link headers.
<i>Wrap</i>	Allowance given for MAC, LLC, or Network layer headers due to protocol wrapping.
<i>Data</i>	Maximum data link layer packet size.
<i>Trail</i>	Sum of the largest MAC and hardware trailers.
<i>Total</i>	Overall size of each packet buffer.
<i>Bytes Alloc</i>	Amount of buffer memory for this device. This value is determined by multiplying the values of <i>Alloc</i> x <i>Total</i> .

## Clear

Resets the statistical information about a network interface from the console display. This command is useful when tracking changes in large counters.

Enter the interface (or net) number as part of the command. To get the interface number, use the GWCON **configuration** command.

**Syntax:** `_clear interface#`

Example: **clear 1**

Clear network statistics? (Yes or No):\_\_

## Configuration

Displays information about the protocols and network interfaces. The output is displayed in three sections, the first section lists the router identification, software version, boot ROM version, and the state of the auto-boot switch. The second and third sections list the protocol and interface information.

**Syntax:** configuration

Example: **configuration**

+conf

```
DECswitch900EF/MP, MP Brouter: 6 Enet 1 Fddi,HW=1/2,#2172,SW=X01.0.000
Hostname: router11
Boot ROM version 0.4
Watchdog timer enabled
Auto-boot switch enabled
Manufacturing rest disabled
Manufacturing test disabled
Console baud rate: 0
```

```
Num Name Protocol
0 IP DOD-IP
3 ARP Address Resolution
11 SNMP Simple Network Management Protocol
```

7 Networks:

Net	Interface	MAC/Data-Link	Hardware	State
0	FDDI/0	FDDI/IEEE 802.2	FDDI	Down
1	Eth/0	Ethernet/IEEE 802.3	Ethernet	Up
2	Eth/1	Ethernet/IEEE 802.3	Ethernet	Up
3	Eth/2	Ethernet/IEEE 802.3	Ethernet	Down
4	Eth/3	Ethernet/IEEE 802.3	Ethernet	Up
5	Eth/4	Ethernet/IEEE 802.3	Ethernet	Down
6	Eth/5	Ethernet/IEEE 802.3	Ethernet	Down

*Protocols:*  
*Num* Indicates the number that is associated with the protocol.

*Name* Indicates the abbreviated name of the protocol.

*Protocol* Indicates the full name of the protocol.

*Networks:*  
*Net* Indicates the network number that the software assigns to the interface.

<i>Interface</i>	Indicates the name of the interface and the connector number on the card that the interface uses.
<i>MAC/Data Link</i>	Indicates the type of MAC/Data link configured for that interface.
<i>Hardware</i>	Indicates the specific kind of interface by hardware type.
<i>State</i>	Indicates the current state of the network interface. The states are: <ul style="list-style-type: none"> <li>• <b>Up</b>           The interface is operational and connected.</li> <li>• <b>Down</b>        The interface is not operational and has failed a self-test.</li> <li>• <b>Disabled</b>    The interface is either temporarily or permanently disabled.</li> <li>• <b>Testing</b>     The interface is undergoing a self-test.</li> <li>• <b>Not installed</b> No interface is present in the router.</li> </ul>

## Disable

Takes a network interface off-line, making the interface unavailable. This command immediately disables the interface. You are not prompted to confirm, and no verification message displays. If you disable an interface with this command, it remains disabled until you use the GWCON **test** command or an OPCON **restart** or **reload** command to enable it.

Enter the interface, or net number as part of the command. To obtain the interface number, use the GWCON **configuration** command.

**Note:** This command may not be accepted during states of interface self-testing. In this case, you receive either or two messages: Network is already undergoing test, cancelling further testing; **or** Network is already undergoing test, cannot disable now. If either message appears, try the **disable** command again later.

**Syntax:** `disable interface#`

Example: `disable 1`

## Error

Displays error statistics for the network. This command provides a group of error counters.

**Syntax:** `error`

Nt	Interface	Input Discards	Input Errors	Input Unk Proto	Input Flow Drop	Output Discards	Output Errors
0	TKR/0	0	0	0	0	0	0
1	TKR/1	0	0	0	0	0	1
2	FDDI/0	0	0	0	0	0	2
3	SL/0	0	0	0	0	0	0
4	SL/1	0	0	0	0	0	0
5	SL/2	0	0	0	0	0	0
6	SL/3	0	0	0	0	0	0
7	Eth/0	0	0	0	0	0	0
8	Eth/1	0	0	0	0	0	0

<i>Nt</i>	Network interface number associated with the software.
<i>Interface</i>	Type of interface.
<i>Input Discards</i>	Number of packets unsuccessfully received.
<i>Input Errors</i>	Number of packets that were found to be defective at the data link.
<i>Input Unk Proto</i>	Number of packets received for an unknown protocol.
<i>Input Flow Drop</i>	Number of packets received that are flow controlled on input.
<i>Output Discards</i>	Number of packets that the router chose to discard rather than transmit due to flow control.
<i>Output Errors</i>	Number of output errors, such as attempts to send over a network that is down or over a network that went down during transmission.

**Note:** The sum of the output discards is not the same as input flow drops over all networks. Output discards may indicate locally originated packets.

## Event

Access the ELS environment to display messages on the console. Refer to Chapter 5 for information about the Event Logging System and its commands. Use the **exit** command to return to the GWCON process.

**Syntax:** `_event`

Example: **event**

```
Event logging system user console
ELS>
```

## Feature

Accesses console commands for specific bridging router features outside of the protocol and network interface console processes.

Enter a question mark after the **feature** command to obtain a listing of the features available for your software release. For example:

```
feature ?
```

To access that feature's console prompt, enter the **feature** command at the GWCON prompt followed by the feature number or short name. Table 4-2 lists available feature numbers and names.

**Table 4-2 Bridging Router Feature Numbers and Names**

<b>Feature Number</b>	<b>Feature Short Name</b>	<b>Accesses the following feature console process</b>
0	WRS	WAN Restoral
1	BRS	Bandwidth Reservation
2	MCF	MAC Filtering

After you access the console prompt for that feature, you may begin entering specific console commands to monitor that feature. To return to the GWCON prompt, enter the **exit** command at the feature's console prompt.

**Syntax:** `feature feature# OR feature-short-name`

Example: `feature 1`  
`feature brs`

```
Reservation console>
```

## Interface

Displays statistical information about the network interfaces (for example, Ethernet, Token-Ring, FDDI). You can use this command without a qualifier to provide a summary of all the interfaces (shown in the output below) or with a qualifier to reveal detailed information of one specific interface.

To obtain the interface number, use the GWCON **configuration** command.

**Syntax:** `interface interface#`

Example: **interface**

Nt	Nt'	Interface	CSR	Vec	Self-Test Passed	Self-Test Failed	Maintenance Failed
0	0	TKR/0	80710000	1	0	0	0
1	1	TKR/1	80710080	1	2	1	0
2	2	FDDI/0	80720000	2	4	2	1
3	3	SL/0	80730000	3	0	0	0
4	4	SL/1	80730000	3	0	0	0
5	5	SL/2	80730000	3	0	230	0
6	6	SL/3	80730000	3	0	0	0
7	7	Eth/0	80740000	4	1	0	0
8	8	Eth/1	80740080	4	1	0	0

**Note:** The screen display varies depending on the device.

<i>Nt</i>	Global interface number.
<i>Nt'</i>	Reserved for dial circuit use
<i>Interface</i>	Interface name.
<i>CSR</i>	Command and Status Register address.
<i>Vec</i>	Interrupt vector.
<i>Self-Test Passed</i>	Number of times self-test succeeded (state of interface changes from down to up).
<i>Self-Test Failed</i>	Number of times self-test failed (state of interface changes from up to down).
<i>Maintenance Failed</i>	Number of maintenance failures.

## Log

View or temporarily change the current logging level of messages that are not included in the Event Logging System.

To display the current logging level, do not enter an octal number as part of the command. To change the logging level, enter the octal number of the new logging level as part of the command. The default logging level is 76 (octal).

**Note:** To change the initial logging level (that is, the level that the router uses when it starts), use the CONFIG **set logging level** command.

**Syntax:** log [*octal\_#*]

Example: log

```
Log lvl: 76
```

## Memory

Displays the current CPU memory usage in bytes, the number of buffers, and the packet sizes.

To use this command, free memory must be available. The number of free packet buffers may drop to zero, resulting in the loss of some incoming packets; however, this does not adversely affect router operations. The number of free buffers should remain constant when the router is idle. If it does not, contact your Customer Service representative.

**Syntax:** memory

Example: **memory**

	Total	Reserve	Never	Perm	Temp	Prev
			Alloc	Alloc	Alloc	Alloc
Heap memory	5463895	201824	5065383	328344	375856	22656
Buffer memory	2097024	12288	14512	2082512		

Number of global buffers: Total = 294, Free = 287, Fair = 57, Low = 58

Global buff size: Data = 4478, Header = 128, Wrap = 92, Trail = 19, Total = 4700

<i>Heap memory:</i>	Memory available for dynamically allocated data structures.
<i>Total</i>	Original amount of space available for allocation for memory (heap size total).
<i>Reserve</i>	Minimum amount of memory needed by the currently configured protocols and features.
<i>Never Alloc</i>	Amount of memory that has never been allocated.
<i>Perm Alloc</i>	Amount of memory requested permanently.
<i>Temp Alloc</i>	Amount of memory allocated temporarily.
<i>Prev Alloc</i>	Amount of memory allocated temporarily and returned.
<i>Buffer memory:</i>	Amount of memory used for buffer space.
<i>Total</i>	Original amount of space available for allocation for buffers.
<i>Reserve</i>	Minimum amount of buffer memory needed by the currently configured protocols and features.
<i>Never Alloc</i>	Amount of buffer memory that has never been allocated.
<i>Perm Alloc</i>	Amount of buffer memory requested permanently.
<i>Number of global buffers:</i>	
<i>Total</i>	Original number of global buffers in the system.
<i>Free</i>	Number of global buffers available.
<i>Fair</i>	Fair number of buffers for each interface. (See <i>Low</i> below.)
<i>Low</i>	<i>Low</i> is the number of free buffers at which the allocation strategy changes to conserve buffers. If the value of <i>Free</i> is less than <i>Low</i> , then buffers are not placed on any queue containing more than the <i>Fair</i> number of buffers on it.

*Global buff  
size:*

<i>Data</i>	Maximum data link packet size of all interfaces.
<i>Header</i>	Sum of the maximum hardware, MAC, and data link headers of all interfaces.
<i>Wrap</i>	Allowance given for MAC, LLC, or Network layer headers due to protocol wrapping.
<i>Trailer</i>	Sum of the largest MAC and hardware trailers of interfaces.
<i>Total</i>	Size of the largest packet buffer.

## Network

Obtains the console prompt for the specified interface and enters the console environment for supported networks, such as X.25 networks. From the prompt, you can display statistical information, such as the routing information fields for X.25 networks.

Enter the interface or net number as part of the command. To obtain the interface number, use the CONFIG **list** command. To return to GWCON, enter **exit**.

**Syntax:** network *interface#*

Example: **network 2**

TKR>

## Protocol

Accesses a protocol's command environment and communicates with the router software that implements the network protocols installed in your router. After entering this command, the prompt of the specified protocol appears. From the prompt, you can enter commands that are specific to that protocol.

Enter the protocol number or short name as part of the command. To obtain the protocol number or short name, enter the CONFIG command environment and enter the **list configuration** command. To return to GWCON, enter **exit**.

**Syntax:** protocol *prot#*

Example: `protocol 7`

`IPX>`

Table 4–3 lists examples of protocol numbers and names.

**Table 4–3 Protocol Numbers and Names**

Protocol Number	Protocol Short Name	Accesses the following protocol process
0	IP	Internet Protocol
3	ARP	Address Resolution Protocol
4	DN	DECnet – a subset of Network Control Program
7	IPX	Novell NetWare Internetwork Packet Exchange
8	OSI	ISO Open Systems Interconnection – Connectionless Network Layer Protocol
9	DVMRP	Distance Vector Multicast Routing Protocol
10	BGR	Border Gateway Protocol
11	SNMP	Simple Network Management Protocol
12	OSPF	Open Shortest Path First
14	APL	AppleTalk Phase 1
20	SDLC	SDLC Relay
22	AP2	AppleTalk Phase 2
23	ASRT	Adaptive Source Routing Transparent Bridge
24	HST	TCP/IP Host Services
26	DLS	Data Link Switching

## Queue

Displays statistics about the length of input and output queues on the specified interfaces. Information about input and output queues provided by the **queue** command includes:

- The total number of buffers allocated

- The low-level buffer value
- The number of buffers currently active on the interface

To display information about one interface only, enter the interface or network number as part of the command. To obtain the interface number, use the **GWCON configuration** command.

**Syntax:** `queue interface#`

Nt	Interface	Input Queue			Output Queue	
		Alloc	Low	Curr	Fair	Curr
0	TKR/0	20	7	0	20	0
1	TKR/1	20	7	20	20	0
2	FDDI/0	40	10	4	30	0
3	SL/0	10	4	0	10	0
4	SL/1	10	4	0	10	0
5	SL/2	10	4	10	10	0
6	SL/3	10	4	0	10	0
7	Eth/0	40	10	40	30	0
8	Eth/1	40	10	40	30	0

*Nt* Network interface number associated with the software.

*Interface* Type of interface.

*Input Queue:*

*Alloc* Number of buffers allocated to this device.

*Low* Low water mark for flow control on this device.

*Curr* Current number of buffers on this device. A value of 0 means the device is disabled.

*Output Queue:*

*Fair* Fair level for the length of the output queue on this device.

*Curr* Number of packets currently waiting to be transmitted on this device. For locally originated packets, the eligibility discard depends on the global low water mark described in the **memory** command.

The router attempts to keep at least the *Low* value packets available for receiving over an interface. If a packet is received and the value of *Curr* is less than *Low*, then the packet is subject to flow control. If a buffer subject to flow control is to be queued on this device and the *Curr* level is greater than *Fair*, then the buffer is

dropped instead of queued. The dropped buffer is displayed in the Output Discards column of the **error** command. It also generates ELS event GW.036 or GW.057.

Due to the scheduling algorithms of the router, the dynamic numbers of *Curr* (particularly the *Input Queue Curr*) may not be fully representative of typical values during packet forwarding. The console code runs when the input queues are drained. Thus, *Input Queue Curr* is non-zero when those packets are waiting on slow transmit queues.

## Statistics

Displays statistical information about the network software, such as the configuration of the networks in the router.

To display information about one interface only, enter the interface or network number as part of the command. To obtain the interface number, use the GWCON **configuration** command.

**Syntax:** `statistics interface#`

Example: **statistics**

Nt	Interface	Unicast Pkts Rcv	Multicast Pkts Rcv	Bytes Received	Packets Trans	Bytes Trans
0	TKR/0	0	0	0	0	0
1	TKR/1	4669	12481	1041127	17150	972535
2	FDDI/0	5269	2843	1176718	10602	1335611
3	SL/0	0	0	0	0	0
4	SL/1	0	0	0	0	0
5	SL/2	0	0	0	230	138000
6	SL/3	0	0	0	0	0
7	Eth/0	3	0	1800	9754	1511069
8	Eth/1	3	0	1800	9751	1509731

*Nt* Network interface number associated with the software.

*Interface* Type of interface.

*Unicast Pkts Rcv* Number of non-multicast, non-broadcast specifically addressed packets at the MAC layer.

*Multicast Pkts Rcv* Number of multicast or broadcast packets received.

<i>Bytes Received</i>	Number of bytes received at this interface at the MAC layer.
<i>Packets Trans</i>	Number of packets transmitted.
<i>Bytes Trans</i>	Number of bytes transmitted at the MAC layer.

## Test

Verifies the state of an interface or enables an interface that was previously disabled with the GWCON **disable** command.

Enter the interface or network number as part of the command. To obtain the interface number, use the GWCON **configuration** command. When testing starts, the console displays a message similar to the following:

```
Testing net 0 FDDI/0...
```

When testing completes, fails, or GWCON times out (after 30 seconds), the console displays a message similar to one of the following:

```
Testing net 0 FDDI/0...successful
```

```
Testing net 0 FDDI/0...failed
```

```
Testing net 0 FDDI/0...still testing
```

Some interfaces may take more than thirty seconds before testing is done.

**Syntax:** `test int_#`

Example: `test 0`

## Uptime

Displays time statistics about the router that includes

- Number of restarts.
- Number of known crashes.
- Whether the router was last reloaded or restarted.
- Time elapsed since the last reload.
- The elapsed since the last restart.

**Syntax:** `uptime`

**Example:** `uptime`

```
1 start, (0 known crashes) Last: Reloaded
Last Reload: 4 hours, 46 minutes ago
Last Restart: 4 hours, 46 minutes ago
```



---

## Using the Event Logging System

This chapter describes the Event Logging System (ELS) and its configuration.

Refer to your hardware installation manual for information about accessing the console for your hardware platform.

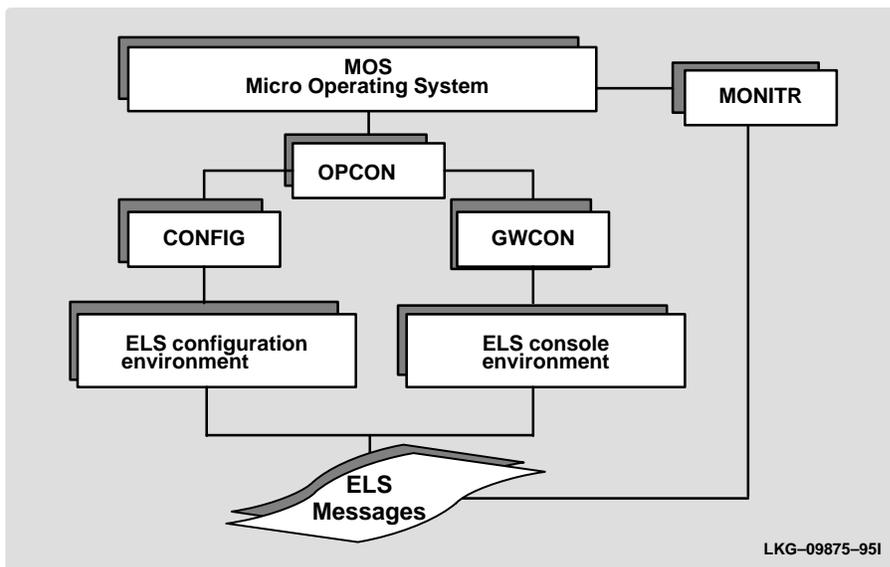
### Using the Event Logging System (ELS)

ELS is a monitoring system that is an integral part of the router operating system. ELS manages the messages logged as a result of router activity. Using ELS commands, you can set up a configuration that sorts out only those messages that are important to you. You can display the messages on the console terminal screen (through the MONITR process), through SNMP (Simple Network Management Protocol), or both. You configure volatile ELS configuration information through the GWCON process; non-volatile ELS configuration through CONFIG.

**Note:** Specific ELS messages are described in the *Event Logging System Messages Guide*.

You access ELS from the OPCON process. Figure 5–1 shows how ELS fits into the router software structure.

**Figure 5–1 ELS in the Router Software Structure**



## Event Logging Concepts

This section describes how events are logged and how to interpret messages. Also described are the concepts of subsystem, event number, and logging level. A large part of ELS functionality is based on commands which take the subsystem, event number, and logging level as parameters.

## Causes of Events

Events occur continuously while the router is operating. They can be caused by any of the following reasons:

- System activity
- Status changes
- Service requests
- Data transmission and reception

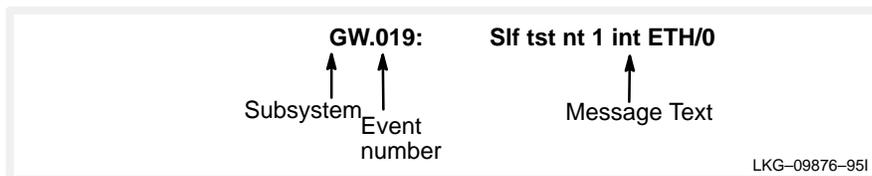
- Data and internal errors

When an event occurs, ELS receives data from the system that identifies the source and nature of the event. Then ELS generates a message that uses the data received as part of the message.

## Interpreting a Message

This section describes how to interpret a message generated by ELS. Figure 5–2 shows the message contents.

**Figure 5–2 Message Generated by an Event**



### Subsystem

Subsystem is a pre-defined short name for a router component, such as a protocol interface. In Figure 5–2, **GW** identifies the subsystem through which this event occurred.

Other examples of subsystems include ARP, IP, TKR, and X25. On a particular router, the actual subsystems present depend on the hardware and software configured for that router. You can use the **list subsystem** command described in this chapter to see a list of the subsystems on your router.

Enter the subsystem as a parameter to an ELS command when you want the command to affect the entire subsystem. For example, the ELS command **display subsystem GW** displays all events that occur through the GW subsystem.

### Event Number

Event Number is a pre-defined, unique, arbitrary number assigned to each message within a subsystem. In Figure 5–2, **19** is the event number within the GW subsystem. To see a list of all the event numbers within a subsystem, issue the **list subsystem** command, where *subsystem* is the short name for the subsystem (for example, GW for GWCON).

The event number always appears with a subsystem, separated by a period. For example: **GW.019**. The subsystem and event number together identify an individual event. They are entered as a parameter to certain ELS commands. When you want a command to affect only the specified event, enter the subsystem and event number as a parameter for the ELS command.

### Message Text

Message Text appears in short form. In Figure 5-2, `Slf tst nt 1 int ETH/0` is the message generated by this event. Variables, such as `source_address` or `network`, are replaced with actual data when the message displays on the console.

The variable `error_code` is referred to by some of the Event Logging System message descriptions (usually preceded by `rsn` or `reason`). They indicate the type of packet error detected. Table 5-1 describes the error or packet completion codes.

**Table 5-1 Packet Completion Codes (Error Codes)**

Code	Meaning
0	Packet successfully queued for output
1	Random, unidentified error
2	Packet not queued for output due to flow control reasons
3	Packet not queued because network is down
4	Packet not queued to avoid looping or bad broadcast
5	Packet not queued because destination host is down (only on networks where this can be detected)

ELS displays network information as follows:

```
nt 1 int Eth/0 (or) network 1, interface Eth/0,
```

- **1** is the network number (each network on the router is numbered sequentially from zero).
- **0** is the unit number (the interfaces of each hardware type are numbered sequentially from zero).

Ethernet and 802.5 hardware addresses appear as a long hexadecimal number.

IP (Internet Protocol) addresses are printed as four decimal bytes separated by periods, such as 18.123.0.16.

DECnet addresses are printed as a pair of decimal numbers separated by a period, such as 2.17.

## Other ELS Terms

This section describes the concepts of logging level and groups as used in messages in the Event Logging System.

### Logging Level

Logging level is a pre-defined setting that classifies each message by the type of event that generated it. This setting is displayed whenever you use the **list subsystem** ELS console command. Table 5–2 lists the logging levels and types.

**Table 5–2 Logging Levels**

<b>Logging Level</b>	<b>Type</b>
<b>UI - ERROR</b>	Unusual internal errors
<b>CI - ERROR</b>	Common internal errors
<b>UE - ERROR</b>	Unusual external errors
<b>CE - ERROR</b>	Common external errors
<b>ERROR</b>	Includes <i>all error</i> levels above
<b>U-INFO</b>	Unusual informational comment
<b>C-INFO</b>	Common informational comment
<b>INFO</b>	Includes <i>all comment</i> levels above
<b>STANDARD</b>	Includes <i>all error</i> levels and <i>all informational comment</i> levels (default)
<b>P-TRACE</b>	Per packet trace
<b>U-TRACE</b>	Unusual operation Trace message
<b>C-TRACE</b>	Common operation Trace message
<b>TRACE</b>	Includes <i>all trace</i> levels above
<b>ALL</b>	Includes <i>all logging levels</i>

In Table 5–2, ERROR, INFO, TRACE, STANDARD, and ALL are aggregates of other logging level types. STANDARD is the recommended default.

The logging level setting affects the operation of the following commands:

- Display subsystem
- Nodisplay subsystem
- Trap subsystem
- Notrap subsystem

The logging level is set for a particular command when you specify it as a parameter to one of the preceding commands. For example:

```
display subsystem TKR ERROR
```

The logging level on the command line modifies the **display** command so that whenever an event with a logging level of either UI-ERROR or CI-ERROR occurs through subsystem TKR, the console displays the resulting message.

You cannot specify the logging level for operations affecting groups or events.

## Groups

Groups are user-defined collections of events that are given a name, the group name. Like the subsystem, subsystem and event number, and logging level, you can use the group name as a parameter to ELS commands. However, there are no pre-defined group names. You must create a group before you can specify its name on the command line.

To create a group, use the **add** configuration command described in this chapter, specify the name you want to call the group, and then specify the events you want to be part of the group. The events you add to the group can be from different subsystems and consists of different logging levels.

After creating a group, you can use the group name to manipulate the events in the group as a whole. For example, to turn off display of all messages from events that have been added to a group named grouptwo, include the group name on the command line, as follows:

```
nodisplay group grouptwo
```

To delete a group, use the **delete** command.

## ELS Configuration Environment

The ELS configuration environment (available at the CONFIG process) is characterized by the `ELS Config>` prompt. Commands entered at this prompt create the ELS default state that take effect after restarting the router. These commands are described in greater detail later in this chapter.

Configuration commands that have subsystem, group, or event as a parameter are executed in the following order:

- Subsystem
- Group
- Event

To set a basic ELS configuration, enter the **display subsystem all standard** command at the `ELS Config>` prompt. This command configures the ELS to display messages from all subsystems with the STANDARD logging level.

**Note:** The default ELS configuration is to display no subsystems. To display ELS messages, you must configure the parameters.

## ELS Console Environment

The ELS console environment (available at the CONFIG process) is characterized by the `ELS>` prompt. Commands entered at this prompt modify the current ELS parameter settings. These commands are described in the next chapter.

## Entering and Exiting ELS

This section describes how to enter and exit the ELS configuration and console environments.

## Entering and Exiting ELS Configuration Environment

To enter the ELS configuration environment from OPCON, perform the following steps:

1. At the OPCON prompt, enter the **status** command to find the pid of CONFIG.

```
* s

Pid   Name   Status  TTY   Comments
  1   COpCon  IOW     TTY0
  2   Monitr  DET     --
  3   Tasker  IDL     --
  4   MOSDDT  DET     --
  5   CGWCon  IOW     --
  6   Config  IOW     --
  7   ROpCon  IOW     TTY1   janb
  8   ROpCon  RDY     TTY2
*
```

2. Enter the **talk** command and the pid number for CONFIG.

```
* talk 6
```

3. The console displays the CONFIG prompt (Config>). If the prompt does not appear when you first enter CONFIG, press **RET**.
4. At the CONFIG prompt, enter the following command to access ELS:

```
Config> eve
```

The console displays the ELS configuration prompt (ELS config>). Now you can enter ELS configuration commands.

To leave the ELS configuration environment, enter the **exit** command. This command is described in this chapter.

## Entering and Exiting ELS Console Environment

To enter the ELS console environment from OPCON, perform the following steps:

1. At the OPCON prompt, enter the **status** command to find the pid of GWCON.

```
* s
```

Pid	Name	Status	TTY	Comments
1	COpCon	IOW	TTY0	
2	Monitr	DET	--	
3	Tasker	IDL	--	
4	MOSDDT	DET	--	
5	CGWCon	IOW	--	
6	Config	IOW	--	
7	ROpCon	IOW	TTY1	janb
8	ROpCon	RDY	TTY2	

\*

2. Enter the **talk** command and the pid number for GWCON.

```
* talk 5
```

3. The console displays the GWCON prompt (+). If the prompt does not appear when you first enter GWCON, press **RET**.
4. At the GWCON prompt, enter the following command to access ELS:

```
+ event
```

The console displays the ELS console prompt (ELS>). Now you can enter ELS console commands. To leave the ELS console environment, enter the **exit** command. This command is described in this chapter.

## ELS Configuration Commands

This section first summarizes the ELS configuration commands and then describes each one in detail. After accessing the ELS configuration environment, you can enter ELS Configuration commands at the `ELS Config>` prompt.

**Table 5–3 ELS Configuration Command Summary**

<b>Command</b>	<b>Function</b>
<b>? (Help)</b>	Lists the ELS configuration commands or lists the options associated with specific commands.
<b>Add</b>	Adds an event to an existing group or creates a new group.
<b>Clear</b>	Clears all ELS configuration information.
<b>Default</b>	Resets the display or trap setting of an event, group, or subsystem.
<b>Delete</b>	Deletes an event number from an existing group or deletes an entire group.
<b>Display</b>	Enables message display on the console monitor.
<b>List</b>	Lists information on ELS settings and messages.
<b>Nodisplay</b>	Disables message display on the console.
<b>Notrap</b>	Keeps messages from being trapped and sent out over SNMP.
<b>Set Pin</b>	Sets the pin parameter to the maximum number of traps per second.
<b>Trap</b>	Allows messages to be trapped and sent out over SNMP.
<b>Exit</b>	Exits ELS configuration and returns you to the CONFIG process.

**? (Help)**

Lists the commands that are available from the current prompt level. You can also enter a ? after a specific command name to list its options.

**Syntax:** ?

Example: ?

```
ADD
CLEAR
DEFAULT
DELETE
DISPLAY
LIST
NODISPLAY
NOTRAP
SET
TRAP
EXIT
```

Example: **list** ?

```
ALL
GROUPS
PIN
STATUS
SUBSYSTEM
```

## Add

Adds an individual event to an existing group creates a new group. Group names must start with an alphabetic character and are case sensitive.

**Syntax:** `add group_name subsystem.event_number`

Example: **add MyGroup gw.019**

**Note:** If the specified group does not exist, the following prompt asks you to confirm the creation of a new group:

```
Group not found. Create new group? (yes or no)
```

## Clear

Clears all ELS configuration information.

**Syntax:** `clear`

Example: **clear**

```
You are about to clear all ELS configuration information
Are you sure you want to do this (Yes or No):
```

## Delete

Deletes an event number from an existing group or deletes the entire group. You are notified, if the specified event is the last event deleted in a group. If you specify *all* instead of *subsystem.event\_number*, a prompt asks you to confirm the deletion of the entire group.

**Syntax:** `delete group_name subsystem.event_number`

Example: `delete groupa gw.019`

## Display

Enables message display on the console monitor for specific events, groups, or subsystems.

**Syntax:** `display event . . .  
group . . .  
subsystem . . .`

### **event** *subsystem.event#*

Displays messages of the specified event (*subsystem.event#*).

Example: `display event gw.019`

### **group** *group.name*

Displays messages of a specified group (*group.name*).

Example: `display group groupb`

### **subsystem** *subsystem.name*

Displays messages associated with the specified subsystem. The following is a list of the supported subsystems of the bridging router.

**Note:** Not all subsystems are supported by all hardware platforms. Refer to your hardware installation guide for information about the specific subsystems that your platform supports.

<b><u>Subsystem</u></b>	<b><u>Description</u></b>
All	All subsystems
GW	Router base and network library
FLT	Filter library
BRS	Bandwidth Reservation
ARP	Address Resolution Protocol
IP	Internet Protocol
ICMP	Internet Control Message Protocol
TCP	Transport Control Protocol
UDP	User Datagram Protocol
BTP	BOOTP relay agent
RIP	IP Routing Information Protocol
EGP	Exterior Gateway Protocol
OSPF	Open SPF-based Routing Protocol
MSPF	OSPF Multicast extensions
TFTP	Trivial File Transfer Protocol
SNMP	Simple Network Management Protocol
DN	DECnet
IPX	Internetwork Packet Exchange Protocol
CLNP	ISO 8473 (Connectionless Layer Network Protocol)
ESIS	ISO 9542 (End-System to Intermediate-System Protocol)
ISIS	ISO 10589 (Intermediate-System to Intermediate-System Protocol)
DNAV	Digital Network Architecture Phase V
APL	AppleTalk
ZIP	AppleTalk Zone Information Protocol
RTMP	AppleTalk Routing Table Management Protocol
AP2	AppleTalk Phase 2
ZIP2	AppleTalk Phase 2 Zone Information Protocol
R2MP	AppleTalk Phase 2 Routing Table Management Protocol
SRT	Source Routing Transparent Bridge
STP	Spanning Tree Protocol
BR	Bridging/Routing
SDLC	SDLC Relay
ETH	Ethernet handler
SL	Proteon Serial Line Handler
TKR	Token Ring Handler
X25	X.25 Protocols
FDDI	FDDI Handler

<b><u>Subsystem</u></b>	<b><u>Description (cont.)</u></b>
FRL	Frame Relay
PPP	Point-to-Point Protocol
X251	X.25 Physical Layer
X252	X.25 Frame Layer
X253	X.25 Packet Layer
IPPN	IP Protocol Net
BGP	Border Gateway Protocol

Example: `display subsystem tkr`

## List

Lists updated information regarding ELS settings and listings of selected messages.

**Syntax:** `list`    all  
                           groups  
                           pin  
                           status  
                           subsystem  
                           subsystem . . .  
                           subsystems all

### all

Lists information from all the **list** categories.

Example: `list all`

### groups

Lists the user-defined group names and contents.

Example: `list groups`

### pin

Lists the current pin value.

Example: `list pin`

## status

Lists the subsystems, groups and events which have been modified by the **display**, **nodisplay**, **trap**, and **notrap** commands.

Example: `list status`

## subsystem

Lists names, events, and descriptions of all subsystems.

Example: `list subsystem`

Name	Events	Description
ALL		All subsystems
GW	31	Router base and network library
ARP	16	Address Resolution Protocol
IP	71	Internet Protocol
ICMP	15	Internet Control Message Protocol
UDP	4	User Datagram Protocol
RIP	19	IP Routing Information Protocol
EGP	53	Exterior Gateway Protocol
OSPF	53	Open SPF-Based Routing Protocol
SNMP	14	Simple Network Management Protocol
DN	84	DECnet
IPX	38	Internetwork Packet Exchange Protocol
ISO	37	ISO Connectionless Network Layer
ESIS	20	ES-IS Information Exchange Protocol
APL	59	AppleTalk
STB	48	Spanning Tree Bridge
ETH	30	Ethernet Handler
SL	15	Proteon Serial Line Handler

## subsystem *subsystem*

Lists all events in a specified subsystem.

Example: `list subsystem gw`

Event	Level	Message
GW.001	ALWAYS	Copyright 1984 Mass Institute of Technology
GW.002	ALWAYS	Portable CGW %s Rel %s strtd
GW.003	ALWAYS	Unus pkt len %d nt %d int %s/%d
GW.004	ALWAYS	Sys %s q adv alloc %d excd %d
GW.005	ALWAYS	Bffrs: %d avail %d idle fair %d low %d
GW.006	C-INFO	Pkt frm nt %d int %s/%d for uninit prt, disc
GW.007	C-INFO	Ip err %x nt %d int %s/%d
GW.008	U-INFO	Ip ovfl nt %d int %s/%d, disc
GW.009	UI-ERROR	Nt dwn ip rstrt nt %d int %s/%d
GW.010	UI-ERROR	Ip q len %d no ip buf nt %d int %s/%d
GW.011	U-INFO	Op err %x hst %wo nt %d int %s/%d
GW.012	U-INFO	Op err cnt excd hst %wo nt %d int %s/%d
GW.013	U-INFO	Rtrns cnt excd hst %wo nt %d int %s/%d
GW.014	UI-ERROR	Nt dwn op rstrt nt %d int %s/%d
GW.015	UI-ERROR	Nt dwn to hst %wo nt %d int %s/%d
GW.016	U-INFO	Op ovfl to hst %wo nt %d int %s/%d
GW.017	UE-ERROR	Intfc hdw mssng nt %d int %s/%d
GW.018	U-TRACE	Strt nt slf tst nt %d int %s/%d
GW.019	C-INFO	Slf tst nt %d int %s/%d
GW.020	U-TRACE	Nt pss slf tst nt %d int %s/%d
GW.021	UE-ERROR	Nt up nt %d int %s/%d
GW.022	U-TRACE	Nt fld slf tst nt %d int %s/%d

### **subsystems all**

Lists all events in all subsystems.

Example: `list subsystems all`

### **Nodisplay**

Use the **nodisplay** command to select and turn off messages displaying on the console monitor.

**Syntax:** `nodisplay event ...`  
`group ...`  
`subsystem ...`

### **event *subsystem.event#***

Suppresses the displaying of a specified event (*subsystem.event#*).

Example: `nodisplay event gw.019`

**group *group.name***

Suppresses the displaying of messages that were previously added to the specified group (*group.name*).

Example: `nodisplay group groupb`

**subsystem *subsystem.name***

Suppresses the display of messages associated with the specified subsystem.

Example: `nodisplay subsystem tkr`

**Notrap**

Selects and turns off messages being trapped and sent over SNMP.

**Syntax:**    `notrap`    `event ...`  
                              `group ...`  
                              `subsystem ...`

**event *subsystem.event#***

Suppresses the trapping of the specified message (*subsystem.event#*).

Example: `notrap event gw.019`

**group *group.name***

Suppresses the trapping messages that were previously added to the specified group (*group.name*).

Example: `notrap group groupb`

**subsystem *logging level***

Suppresses the trapping of messages associated with the specified subsystem (*logging level*).

Example: `notrap subsystem tkr error`

**Set Pin**

Sets the pin parameter to the maximum number of traps per second that can be sent. Internally, the pin resets every tenth of a second. One-tenth of the number *max\_traps* is sent every tenth of a second.

**Syntax:**    `set pin max_traps`

Example: `set pin 100`

## Trap

Selects and turns on messages trapped and sent over SNMP.

**Syntax:**    trap        event . . .  
                              group . . .  
                              subsystem . . .

### **event** *subsystem.event#*

Traps the specified message (*subsystem.event#*).

Example: `trap event gw.019`

### **group** *group.name*

Traps messages that were previously added to the specified group.

Example: `trap group groupb`

### **subsystem** *logging level*

Traps messages associated with the specified subsystem (*logging level*).

Example: `trap subsystem gw`

## Exit

Exits ELS and returns to the CONFIG prompt.

**Syntax:**    exit

Example: `exit`

The next section describes how to use ELS to monitor and troubleshoot the network.

## Monitoring the Event Logging System

This section describes how to monitor events logged by ELS and how to use the ELS console commands. The information includes the following sections:

- Accessing the ELS Console Environment
- ELS Console Commands

### Process Characteristics

When you view ELS from the MONitor process, you will see a considerable amount of information. Because the router cannot buffer and display every packet under moderate to heavy loads, the buffers are flushed. When this occurs, the following message appears on the screen:

```
xx messages flushed
```

The bridging router provides no functionality to save these messages so that they can be reviewed later. When this message appears, you may want to tailor the ELS output to display information that is important to the current task you are monitoring.

It is also important to note that the ELS messages continually rotate through the router's buffers. To stop and restart the ELS message use

<code>CTRL/s</code>	to stop the scrolling
<code>CTRL/q</code>	to start scrolling

You can also capture the ELS output to a file. To do this, start a script file or log file from your location when telneting to a router. You can also do this by attaching a PC to the router's console port and starting a log file from within the terminal emulation package. Customer Service may need this information to diagnose a problem.

### Using the ELS Monitor Process Effectively

To use ELS effectively, follow these suggestions.

- Know what you want to see before using the ELS system. Clearly define the problem or events that you want to see before using the MONitor process.

- Execute the command **nodisplay subsystem all all** to turn off all ELS messages.
- Turn on only those messages that relate to the problem you are experiencing.
- Use the *Event Logging System Messages Guide* to determine which messages are normal.

## Using ELS to Troubleshoot a Problem

When you are trying to troubleshoot a particular problem, display only those messages that relate to the problem. If, for example, you are experiencing a problem with bridging, turn on the bridging messages by issuing the following commands:

```
display subsystem srt all  
display subsystem br all
```

Initially, because of the speed at which the messages scroll off the screen, you may want to record the numbers and look them later up in the manual. After you become familiar with the different types of messages, you can selectively turn on and off those messages you need to troubleshoot a problem. The following sections list specific ELS examples. Different problems may require different steps.

### ELS Example 1

You are interested into looking at the frequency of polling on a token ring interface, and whether the polls are successful.

```
ELS> nodisplay subsystem all all
```

```
ELS> display subsystem tkr all
```

```
CTRL/p
```

```
* t 2
```

As the messages begin to scroll, look for ELS message tkr.031.

## ELS Example 2

The SR-TB bridging is not working.

1. Check the configuration.
2. Use the GWCON bridging console to verify that the bridging interfaces are enabled.
3. Enter the following:

```
* t 6
config> event
ELS config> nodisplay subsystem all all
ELS config> display subsystem srt all
ELS config> exit
config> 
* restart
* T 2
```

As the message begin to scroll, look for messages srt.071 through srt.075. If you see one these messages, you are not licensed to use one or more of the bridging features.

## ELS Example 3

The router cannot communicate with an IPX server on an ethernet.

1. Using the IPX console (GWCON) command, issue the **SLIST** command to verify that the server is listed.
2. Check the IPX configuration.
3. Enter the following:

```
* t 5
+ event
ELS> nodisplay subsystem all all
ELS> display subsystem IPX all
ELS> display subsystem eth all
ELS> 
* t 2
```

As the messages begin to scroll, look for ELS message eth.006. This indicates that the server has a bad Ethernet config.

## Accessing the ELS Console Environment

For information on how to access the ELS console environment, see the section “ELS Configuration Environment” in this chapter.

## ELS Console Commands

This section summarizes the ELS console commands. After accessing the ELS Console environment, you can enter ELS console commands at the `ELS>` prompt.

**Table 5–4 ELS Console Command Summary**

Command	Function
<b>? (Help)</b>	Lists the ELS console commands or lists the options associated with specific commands.
<b>Clear</b>	Clears messages associated with specific events, groups, or subsystems.
<b>Display</b>	Enables message display on the console.
<b>Exit</b>	Exits ELS console and returns the user to the GWCON process.
<b>List</b>	Lists information on ELS settings and messages.
<b>Nodisplay</b>	Disables message display on the console.
<b>Notrap</b>	Keeps messages from being trapped and sent out over SNMP.
<b>Remove</b>	Frees up memory by erasing stored information.
<b>Restore</b>	Clears current settings and reloads initial ELS configuration.
<b>Retrieve</b>	Reloads the saved ELS configuration.
<b>Save</b>	Stores the current configuration.
<b>Set</b>	Sets the pin parameter to the maximum number of traps per second.
<b>Statistics</b>	Displays available subsystems and pertinent statistics.
<b>Trap</b>	Allows messages to be trapped and sent out over SNMP.

## ? (Help)

Lists the commands that are available from the current prompt level. You can also enter a ? after a specific command name to list its options.

**Syntax:** ?

Example: ?

```
CLEAR
DISPLAY
LIST
NODISPLAY
NOTRAP
REMOVE saved state
RESTORE initial state
SAVE current state
SET
STATISTICS
TRAP
EXIT
```

Example: **list** ?

```
ALL
ACTIVE
EVENT
GROUPS
PIN
SUBSYSTEMS
```

## Clear

Disables both the **display** and **trap** commands as they relate to specific events, groups, or subsystems.

**Syntax:** clear event ...  
group ...  
subsystem ...

**event** *subsystem.event#*

Disables displaying or trapping messages for the specified event (*subsystem.event#*).

Example: **clear event gw.019**

**group *group.name***

Disables displaying or trapping messages for the specified group (*group.name*).

Example: `clear group groupb`

**subsystem *subsystem.name***

Disables displaying or trapping messages associated with the specified subsystem (*logging level*). If you do not specify a logging level, all messages for that subsystem are disabled.

Example: `clear subsystem gw`

**Display**

Enables the message display of specified events on the console monitor.

**Syntax:**    `display` event . . .  
                  group . . .  
                  subsystem . . .

**event *subsystem.event#***

Displays messages for the specified event (*subsystem.event#*).

Example: `display event gw.019`

**group *group.name***

Displays the messages that were previously added to the specified group (*group.name*).

Example: `display group groupb`

**subsystem *subsystem.name***

Displays any messages associated with the specified subsystem (*logging level*). If you do not specify a logging level, all messages for that subsystem are turned on.

Example: `display subsystem tkr`

## Exit

Exits ELS and returns to the previous prompt level.

**Syntax:** `exit`

Example: `exit`

## List

Updates information regarding ELS settings and displays listings of selected messages.

**Syntax:** `list` all  
`active . . .`  
`event . . .`  
`groups . . .`  
`pin`  
`subsystem . . .`

### all

Displays information from all the **list** categories.

Example: `list all`

### active *subsystem.name*

Displays the events that are active for a specific subsystem and the count of the occurrence of the messages.

Example: `list active ip`

Event	Active	Count
IP.007		89354
ETH.009	D	10

Subsystem X25: no event active

### event *event.name*

Displays the logging level, the message, and the count of the specified event.

Example: `list event ip.007`

```
Level: p-TRACE
Message: %I -> %I
Active: Count: 84182
```

### **groups *group.name***

Displays the user-defined group names.

Example: `list groups`

### **pin**

Displays the current pin value.

Example: `list pin`

### **subsystem *subsystem.name***

Lists event names, the total number of events that occurred, and their descriptions as shown in the following example.

`list subsystem`

Name	Events	Description
All		All subsystems
GW	53	Router base and network library
FLT	7	Filter library
BRS	4	Bandwidth Reservation
ARP	16	Address Resolution Protocol
IP	81	Internet Protocol
ICMP	19	Internet Control Message Protocol
TCP	48	Transport Control Protocol
UDP	4	User Datagram Protocol
BTP	5	BOOTP relay agent
RIP	19	IP Routing Information Protocol
EGP	54	Exterior Gateway Protocol
OSPF	56	Open SPF-based Routing Protocol
MSPF	14	OSPF Multicast extensions
TFTP	24	Trivial File Transfer Protocol
SNMP	20	Simple Network Management Protocol

### **subsystem *subsystem.name***

Lists all events, logging levels, and messages for the specified subsystem. See the following example.

```
list subsystem eth
```

Event	Level	Message
ETH.001	P-TRACE	brd rcv unkwn type %X %E -> %E nt %d int %s/%d
ETH.002	UE-ERROR	rcv unkwn typ %X %E -> %E nt %d int %s/%d
ETH.010	C-INFO	LLC unk SAP %X %E -> %E nt %d int %s/%d

### subsystems all

Lists all events, logging levels, and messages for every event that occurred on the bridging router. See the following example.

```
list subsystems all
```

## Nodisplay

Selects and turns off the display of messages on the console monitor.

**Syntax:** `nodisplay event . . .`  
`group . . .`  
`subsystem . . .`

### event *subsystem.event#*

Suppresses the display of messages for the specified event as shown in the following example.

```
nodisplay event gw.019
```

### group *group.name*

Suppresses the display of messages that were previously added to the specified group (*group.name*) as shown in the following example.

```
nodisplay group groupb
```

### subsystem *subsystem.name*

Suppresses the display of messages associated with the specified subsystem (*logging level*) as shown in the following example.

```
nodisplay subsystem tkr
```

## Notrap

Selects and turns off messages trapped and sent over SNMP.

**Syntax:**    notrap    event . . .  
                                  group . . .  
                                  subsystem . . .

### **event** *subsystem.event#*

Suppresses the trapping of messages for the specified event (*subsystem#*).

```
notrap event gw.019
```

### **group** *group.name*

Suppresses trapping of messages that were previously added to the specified group (*group.name*).

```
notrap group groupb
```

### **subsystem** *logging level*

Suppresses trapping of messages associated with the specified subsystem (*logging level*).

```
notrap subsystem tkr error
```

## Remove

Recovers memory by erasing stored information. If you have previously saved the current configuration with the **save** command, **remove** allows you to erase the saved configuration.

**Syntax:**    remove

Example: **remove**

## Restore

Clears all current settings (except counters) and reloads the initial ELS configuration. To retain the current settings, use the **save** command before restoring the initial configuration.

**Syntax:**    restore

Example: **restore**

## Retrieve

Reloads the saved ELS configuration. If you have previously saved the current configuration with the **save** command, use **retrieve** to reload it. **Retrieve** does not erase the saved configuration after it executes. To erase the saved configuration, use the **remove** command.

**Syntax:** `retrieve`

Example: `retrieve`

## Save

Stores the current configuration (except counters). **Save** does not affect the default configuration (the one you set with the configuration commands). Use **save** after modifying the configuration with the console commands with the intentions of saving this configuration over a restart. There can be only one saved configuration at a time. To reload the saved configuration, use the **retrieve** command.

**Note:** A saved configuration from one release may not be valid (or the same) when retrieved while running a newer release.

**Syntax:** `save`

Example: `save`

## Set Pin

Sets the pin parameter to the maximum number of traps per second that can be sent. Internally, the pin resets every tenth of a second. One tenth of the number *max\_traps* is sent every tenth of a second.

**Syntax:** `set pin max_traps`

Example: `set pin 100`

## Statistics

Displays a list of the available subsystems and their statistics.

**Syntax:** `statistics`

Example: **statistics**

Subsys	Vector	Exist	String	Active	Heap
GW	50	31	1033	3	34
ARP	25	16	483	0	0
IP	80	70	1686	11	22
ICMP	20	14	335	0	0
UDP	10	4	109	0	0
RIP	30	19	402	0	0
EGP	70	53	1088	0	0
OSPF	60	51	1868	0	0
SNMP	20	12	311	0	0
DN	90	73	4056	0	0
APL	60	59	68	0	0
ETH	16	16	591	0	0
PN	20	15	708	0	0
SL	15	12	393	0	0
TKR	20	19	834	0	0
X25	50	38	1324	1	30
Total	636	502	15289	15	86

Maximum: 1137 vector, 90 subsystem

Memory: 10233/360 vector + 8480/15696 data + 86 heap = 34855

<i>Subsys</i>	Name of subsystem
<i>Vector</i>	Maximum size of subsystem
<i>Exist</i>	Number of events defined in this subsystem
<i>String</i>	Number of bytes used for message storage in this subsystem
<i>Active</i>	Number of active (displayed, trapped, or counted) events in the subsystem
<i>Heap</i>	Dynamic memory in use by subsystem

## Trap

Selects and turns on messages trapped and sent over SNMP.

**Syntax:**   trap        event . . .  
                                  group . . .  
                                  subsystem . . .

### **event** *subsystem.event#*

Traps the specified message (*subsystem.event#*).

Example: `trap event gw.019`

### **group** *group.name*

Traps messages that were previously added to the specified group.

Example: `trap group groupb`

### **subsystem** *logging level*

Traps messages associated with the specified subsystem (*logging level*).

Example: `trap subsystem gw`

---

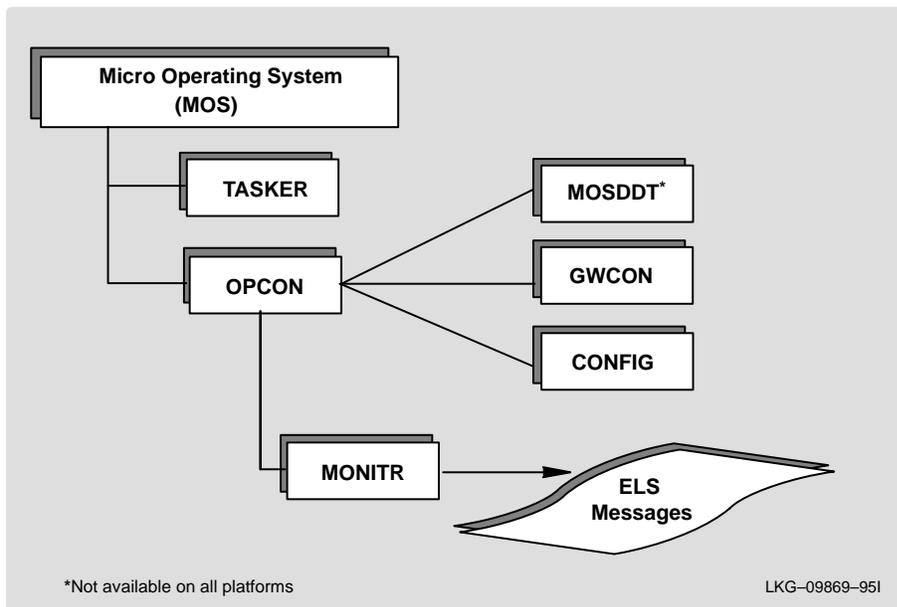
## The MONITR Process

This chapter explains how to use the MONITR process and how to control the way MONITR collects and displays messages.

### The MONITR Process

The MONITR process receives messages from the Event Logging System (ELS) and displays them on the console monitor. MONITR also displays logging messages from software that still uses the old logging system. MONITR fits into the router software structure as shown in Figure A-1.

Figure A-1 MONITR in the Router Software Structure



## Commands Affecting MONITR

The following commands affect the MONITR process:

- OPCON commands:
  - **divert** temporarily sends output to a different device.
  - **flush** causes MONITR to discard the messages it collects.
  - **halt** reverses the action of the **divert** command.
  - **talk** causes MONITR to display its output.
- The CONFIG **set logging disposition** command sets the initial device to which MONITR sends its output.

## Entering and Exiting MONITR

To enter the MONITR process from OPCON, perform the following steps:

1. At the OPCON prompt, enter the **status** command to find the pid of MONITR.

```
* s

Pid   Name   Status  TTY   Comments
  1   COpCon  IOW     TTY0
  2   Monitr  DET     --
  3   Tasker  IDL     --
  4   MOSDDT  DET     --
  5   CGWCon  IOW     --
  6   Config  IOW     --
  7   ROpCon  IOW     TTY1   janb
  8   ROpCon  RDY     TTY2
```

2. Enter the **talk** command and the pid number to enter the MONITR environment.

```
* talk 2
```

MONITR does not display a prompt and you cannot enter any commands. However, the console begins to display the messages MONITR has accumulated.

To exit MONITR and return to OPCON, enter the OPCON intercept character CTRL/p.

## Receiving MONITR Messages

To receive MONITR messages at your console, enter the MONITR process as described in the previous section. MONITR displays all the messages it recorded since it was last invoked. While you are connected to MONITR, it displays all messages as they arrive.

You can use the OPCON **divert** and **halt** commands to view MONITR messages while you are performing another task with the router. Permitted devices divert output to TTY0 (the local console), TTY1, or TTY2 (the remote consoles).

To specify a default device for MONITR, define the device in NVRAM by using the CONFIG **set logging disposition** command. Specifying a default device is useful if you have a terminal set up to print.

---

# Index

## Symbols

?(Help)

- Boot CONFIG command, 3–7
- CONFIG command, 2–8
- ELS configuration command, 5–11
- ELS console command, 5–24
- GWCON command, 4–5
- OPCON command, 1–9

## A

Add

- Boot CONFIG command, 3–7
- CONFIG command, 2–9
- ELS configuration command, 5–12

Address entries

- changing, 3–9
- deleting, 3–12

## B

- Boot, GWCON command, 4–5
- Boot and dump configuration database, displaying, 3–15

Boot CONFIG commands

- ?(Help), 3–7
- add, 3–7
- change, 3–9
- delete, 3–12
- disable, 3–13
- enable, 3–14
- exit, 3–15
- list, 3–15
- load, 3–16
- summary of, 3–6
- TFTP, 3–19

Boot CONFIG process

- commands available from, 3–6
- description of, 3–1
- entering and exiting, 3–2
- entering from CONFIG, 2–10
- exiting, 3–15

Boot directory, 3–4

Boot file, description of, 3–2

Booting, BootP forwarding utility, 3–23

BOOTP

- enabling/disabling, 3–24
- server, 3–25

BootP Forwarding Process, router as  
BootP Client, 3–23

BootP Forwarding Utility, 3–23  
Bootstrap monitor, forwarding process,  
3–23  
Bootstrap protocol, 3–23  
Breakpoint, OPCON command, 1–10  
Buffer, GWCON command, 4–6

## C

CGWCON commands, 4–3  
CGWCON process  
description of, 4–1  
entering and exiting, 4–2  
Change, Boot CONFIG command, 3–9  
Clear  
CONFIG command, 2–12  
ELS configuration command, 5–12  
ELS console command, 5–24  
GWCON command, 4–7  
clock, setting and changing, 2–27  
CONFIG commands  
?(Help), 2–8  
add, 2–9  
boot, 2–10, 2–11  
change, 2–11  
clear, 2–12  
delete, 2–13  
disable, 2–14  
enable, 2–15  
event, 2–16  
feature, 2–16  
list, 2–17  
network, 2–19  
patch, 2–20  
protocol, 2–21  
quick config, 2–22  
set, 2–23

summary of, 2–6  
time, 2–27  
unpatch, 2–28  
update, 2–28  
CONFIG process  
commands available from, 2–6  
description of, 2–1  
entering, 1–19, 2–5  
exiting, 2–5  
CONFIG-ONLY mode, changes to, 2–1  
Configuration  
GWCON command, 4–8  
network interfaces, 1–20  
updating memory, 2–28  
Configuration command, 1–25  
Configuring, user access, 2–5  
CPU, displaying memory usage of, 4–14

## D

date, setting and changing, 2–27  
DDT debugger, entering, 1–10  
Delete  
Boot CONFIG command, 3–12  
CONFIG command, 2–13  
ELS configuration command, 5–13  
Directories, boot and dump, 3–4  
Disable  
Boot CONFIG command, 3–13  
CONFIG command, 2–14  
GWCON command, 4–9  
Display  
ELS configuration command, 5–13  
ELS console command, 5–25  
Divert, OPCON command, 1–10  
Dump file, description of, 3–4  
dumping  
configuring for, 3–3

- dump directories, 3–4
- dump files, 3–4

## E

### ELS

- concepts of, 5–2
- configuration commands, 5–10
- configuration environment, 5–8
- console environment, 5–8
- description of, 5–1
- entering, 2–16
- entering and exiting, 5–8
- exiting to CONFIG prompt, 5–19
- exiting to GWCON prompt, 5–26
- interpreting messages, 5–3
- monitoring, 5–20
- reloading, 5–30
- storing, 5–30
- trapping, 5–32
- troubleshooting example 1, 5–21
- troubleshooting example 2, 5–22
- troubleshooting example 3, 5–22
- using to troubleshoot, 5–21

### ELS configuration commands

- ?(Help), 5–11
- add, 5–12
- clear, 5–12
- delete, 5–13
- display, 5–13
- exit, 5–19
- list, 5–15
- nodisplay, 5–17
- notrap, 5–18
- set pin, 5–18
- summary of, 5–10
- trap, 5–19

### ELS console commands, 5–23

- ?(Help), 5–24
- clear, 5–24
- display, 5–25
- exit, 5–26
- list, 5–26
- nodisplay, 5–28
- notrap, 5–29
- remove, 5–29
- restore, 5–29
- retrieve, 5–30
- save, 5–30
- set pin, 5–30
- statistics, 5–30
- trap, 5–32

### ELS messages

- error codes, 5–4
- explanation, 5–4
- groups, 5–7
- interpreting, 5–3
- logging level, 5–5
- network information, 5–4
- suppressing display of, 5–17, 5–28
- suppressing trapping of, 5–18, 5–29
- trapping, 5–19

### Enable

- Boot CONFIG command, 3–14
- CONFIG command, 2–15

### Entering, protocol configuration, 1–21

### Error, GWCON command, 4–10

### Error codes, ELS Messages, 5–4

### Event

- CONFIG command, 2–16
- GWCON command, 4–11

### Event logging, subsystem, 5–3

### Event number parameter, 5–3

## Exit

- Boot CONFIG command, 3–15
- ELS configuration command, 5–19
- ELS console command, 5–26

## Exiting

- network interface configuration process, 1–21
- protocol configuration, 1–22
- protocol console process, 1–26

## F

### Feature

- CONFIG command, 2–16
- GWCON command, 4–11

### Features

- accessing configuration and console prompts, 1–27
- bandwidth reservation, 2–16, 4–11
- BOSS SNMP Agent, 2–16, 4–11
- MAC filtering, 2–16, 4–11
- WAN restoral, 2–16, 4–11

### Files, boot, 3–2

### Flow control, packets, 4–7, 4–18

### Flush, OPCON command, 1–11

### Forwarding process, example, 3–24

## G

### Group

- adding, 5–12
- deleting, 5–13

### Group name parameter, 5–7

### GWCON commands

- ?(Help), 4–5
- boot, 4–5
- buffer, 4–6

### clear, 4–7

### configuration, 4–8

### disable, 4–9

### error, 4–10

### event, 4–11

### feature, 4–11

### interface, 4–12

### log, 4–13

### memory, 4–14

### network, 4–16

### protocol, 4–16

### queue, 4–17

### statistics, 4–19

### summary of, 4–3

### test, 4–20

### uptime, 4–21

### GWCON process

#### description of, 4–1

#### entering and exiting, 4–2

## H

### Halt, OPCON command, 1–11

### Help command, router user interface, 1–7

## I

### Intercept, OPCON command, 1–11

### Intercept character, changing, 1–11

### Interface, GWCON command, 4–12

### Interface console process, exiting, 1–25

### Interface device

- adding, 2–9
- changing, 2–11

### IP

#### Disabling BOOTP forwarding, 3–24

#### Enabling BOOTP forwarding, 3–24

TFTP, 3–17

## L

### List

Boot CONFIG command, 3–15

CONFIG command, 2–17

ELS configuration command, 5–15

ELS console command, 5–26

List configuration command, 1–21

List Devices command, 1–19

Load, Boot CONFIG command, 3–16

Local console, 1–2

Log, GWCON command, 4–13

### Logging in

from remote console, 1–3

remote login name, 1–2

### Logging level

changing, 4–13

parameter, 5–5

viewing, 4–13

### Login

disabling, 2–14

enabling, 2–15

Logout, OPCON command, 1–12

## M

### Memory

displaying information about, 4–14

erasing information, 5–29

GWCON command, 4–14

obtaining information about, 1–12

OPCON command, 1–12

### Memory dump

disabling, 3–13

enabling, 3–14

### Modem

disabling, 2–14

enabling, 2–15

Monitoring, network interfaces, 1–25

### MONITR process

commands affecting, A–2

description of, A–1

entering and exiting, A–3

OPCON commands, A–2

receiving messages, A–3

## N

### Network

CONFIG command, 2–19

environment, 2–19, 4–16

GWCON command, 4–16

Network command, 1–20

### Network interface

deleting, 2–13

disabling, 4–9

displaying information about, 2–17,  
4–8, 4–12

displaying the configuration, 1–20

enabling, 4–20

verifying, 4–20

Network software, displaying statistical  
information about, 4–19

### Nodisplay

ELS configuration command, 5–17

ELS console command, 5–28

### Notrap

ELS configuration command, 5–18

ELS console command, 5–29

## O

### OPCON commands, A-2

- ?(Help), 1-9
- breakpoint, 1-10
- divert, 1-10
- flush, 1-11
- halt, 1-11
- intercept, 1-11
- logout, 1-12
- memory, 1-12
- reload, 1-13
- restart, 1-12
- status, 1-13
- summary of, 1-8
- talk, 1-15
- telnet, 1-15

### OPCON process

- commands available from, 1-8
- description of, 1-6
- entering, 1-7
- getting back to, 1-17

### Output

- discarding, 1-11
- sending to other consoles, 1-10
- suspending, 1-11

## P

### Packet completion codes, 5-4

### Packet Forwarder, entering CONFIG environment for, 2-21

### Parameters

- configuring, 2-23
- event number, 5-3
- group name, 5-7
- logging level, 5-5
- subsystem, 5-3

### Passwords, 1-2

### Patch, CONFIG command, 2-20

### Pin parameter, setting, 5-18, 5-30

### Prompts

- CONFIG, 1-7
- GWCON, 1-7
- OPCON, 1-7
- router processes, 1-7

### Protocol

- CONFIG command, 2-21
- configuration process, 1-17
- console process, 1-23
- GWCON command, 4-16
- IDs, 1-26
- names and numbers, 1-26

### Protocol command, 1-22, 1-25

### Protocol console process

- entering, 1-24, 1-25
- exiting, 1-26

### Protocols

- console process, 1-23
- displaying information about, 4-8
- entering configuration environment for, 2-21

## Q

### Queue, GWCON command, 4-17

### Quick config, CONFIG command, 2-22

## R

### Reload, OPCON command, 1-13

### Remote consoles, 1-2

### Remote login, 1-3

### Remove, ELS console command, 5-29

### Restart, OPCON command, 1-12, 1-23

### Restarting, router, 1-23

Restore, ELS console command, 5–29  
Retrieve, ELS console command, 5–30  
Router  
  delete configuration information, 2–12  
  displaying information about, 2–17  
  OPCON command, 1–12  
  rebooting, 1–13  
Router console, local, 1–2  
Router consoles  
  remote, 1–2  
  using, 1–1  
Router interface  
  communicating with processes, 1–8  
  list of processes, 1–4  
  user, 1–4  
Router processes  
  attaching to, 1–15  
  communicating with, 1–4  
  connecting to, 1–8  
  displaying information about, 1–13  
  list of, 1–4  
Router software  
  communicating with, 4–16  
  reloading, 1–13  
  user interface, 1–1, 1–4

## **S**

Save, ELS console command, 5–30  
Session, terminating, 1–12  
Set, CONFIG command, 2–23  
Set Password command, 1–2  
Set Pin  
  ELS configuration command, 5–18  
  ELS console command, 5–30  
SRAM device records  
  recreating, 2–9

  replacing, 2–11  
Statistical information, clearing, 4–7  
Statistics  
  ELS console command, 5–30  
  GWCON command, 4–19  
Status, OPCON command, 1–13  
Subsystem parameter, 5–3

## **T**

Talk, OPCON command, 1–15  
Telnet  
  closing a connection, 1–17  
  closing a telnet session, 1–17  
  description of, 1–15  
  obtaining status of telnet session,  
    1–16, 1–17  
  quitting a session, 1–17  
  quitting a telnet session, 1–17  
  telnet command, 1–16  
Telnet connections, 1–2  
Test, GWCON command, 4–20  
TFTP  
  Boot CONFIG command, 3–19  
  description of, 3–17  
  messages, 3–21  
time, setting and changing, 2–27  
Trap  
  ELS configuration command, 5–19  
  ELS console command, 5–32

## **U**

Unpatch, CONFIG command, 2–28  
Update, CONFIG command, 2–28  
Uptime, GWCON command, 4–21  
User access  
  adding user, 2–10

- changing password, 2–11
- changing user, 2–11
- deleting user, 2–14
- listing user information, 2–19

- removing user access, 2–13
- User interface
  - processes, 1–4
  - router software, 1–4

## HOW TO ORDER ADDITIONAL DOCUMENTATION

### DIRECT TELEPHONE ORDERS

In Continental USA  
call 800-DIGITAL

In Canada  
call 800-267-6215

In New Hampshire  
Alaska or Hawaii  
call 603-884-6660

In Puerto Rico  
call 809-754-7575 x2012

### ELECTRONIC ORDERS (U.S. ONLY)

Dial 800-234-1998 with any VT100 or VT200  
compatible terminal and a 1200 baud modem.  
If you need assistance, call 1-800-DIGITAL.

### DIRECT MAIL ORDERS (U.S. AND PUERTO RICO\*)

U. S. SOFTWARE SUPPLY BUSINESS  
DIGITAL EQUIPMENT CORPORATION  
10 Cotton Road  
Nashua, New Hampshire 03063-1260

### DIRECT MAIL ORDERS (Canada)

DIGITAL EQUIPMENT OF CANADA LTD.  
940 Belfast Road  
Ottawa, Ontario, Canada K1G 4C2  
Attn: A&SG Business Manager

### INTERNATIONAL

DIGITAL  
EQUIPMENT CORPORATION  
A&SG Business Manager  
c/o Digital's local subsidiary  
or approved distributor

Internal orders should be placed through the Software Services Business (SSB)  
Digital Equipment Corporation, Westminister, Massachusetts 01473

\*Any prepaid order from Puerto Rico must be placed  
with the Local Digital Subsidiary:  
809-754-7575 x2012



**READER'S COMMENTS**

What do you think of this manual? Your comments and suggestions will help us to improve the quality and usefulness of our publications.

Please rate this manual:

	Poor			Excellent	
Accuracy	1	2	3	4	5
Readability	1	2	3	4	5
Examples	1	2	3	4	5
Organization	1	2	3	4	5
Completeness	1	2	3	4	5

Did you find errors in this manual? If so, please specify the error(s) and page number(s).

---

---

---

---

General comments:

---

---

---

---

Suggestions for improvement:

---

---

---

---

Name \_\_\_\_\_ Date \_\_\_\_\_

Title \_\_\_\_\_ Department \_\_\_\_\_

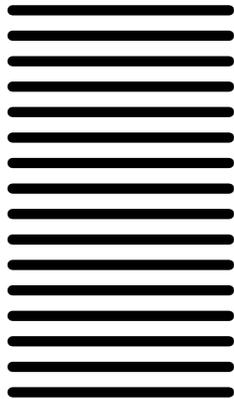
Company \_\_\_\_\_ Street \_\_\_\_\_

City \_\_\_\_\_ State/Country \_\_\_\_\_ Zip Code \_\_\_\_\_

DO NOT CUT – FOLD HERE AND TAPE



NO POSTAGE  
NECESSARY  
IF MAILED  
IN THE  
UNITED STATES



**BUSINESS REPLY LABEL**  
FIRST CLASS PERMIT NO. 33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE



**Shared Engineering Services**

550 King Street  
Littleton, MA 01460–1289

DO NOT CUT – FOLD HERE