IRIS Series 3000
Owner's Guide

Version 2.0

Silicon Graphics, Inc.
2011 Stierlin Road
Mountain View, CA 94043

Document Number 007-5220-020

[This page missing from original copy]

CONTENTS

List of Figures

List of Tables

1. Introduction

This document explains how to install and test IRIS series 3000 products, including the series 3100 packages and the series 2000 Turbo products. Read this document carefully before installing your IRIS.

1.1 Installation Overview

The procedures for installing IRIS series 3000 products include:

- Installing hardware (Chapter 2)
- Booting an IRIS workstation (Chapter 3)
- Booting an IRIS terminal (Chapter 3)
- Making a bootable backup tape for the IRIS workstation (Chapter 4)
- Configuring software for the IRIS workstation (Chapter 4)
- Configuring software for the IRIS terminal (Chapter 5)
- Configuring a disk (Chapter 6)
- Installing optional peripherals, including ASCII terminals (Chapter 7)
- Installing non-standard video monitors (Chapter 8)

For basic information about UNIX commands, the UNIX text editor, and navigating the file system, see the pamphlet entitled *Getting Started with Your IRIS Workstation*.

1.2 IRIS Series 2000 Turbo Products

You can use this manual with the IRIS 2300T (3010 equivalent), 2400T (3020 equivalent), and 2500T. Early IRIS series 2000 models with the Turbo upgrade may not have the same standard peripherals described here. The 60Hz monitor may be different from the one described in this guide. The mouse may be mechanical rather than optical. The early IRIS 2400 Turbo system required a separate junction box. For a description of these features, see the *IRIS Workstation Guide Series 2000, Version 1.0*.

1.3 IRIS Series 3100 Packages

The IRIS series 3100 products consist of the following packages:

- 3110–a standard IRIS 3010, with Z-clipping and 10-MHz Geometry Engines. You can use all the documentation (e.g., system administration, software, and hardware configuration) for the 3010 for the 3110.

- 3120–a standard IRIS 3020, with Z-clipping and 10-MHz Geometry Engines. You can use all the documentation (e.g., system administration, software, and hardware configuration) for the 3020 for the 3120.

- 3130–a standard IRIS 3030, with a floating point board, a cartridge tape drive, 32 bitplanes, 8 megaytes of RAM, Z-clipping and 10-MHz Geometry Engines. You can use all the documentation (e.g., system administration, software, and hardware configuration) for the 3030 for the 3130.

1.4 Product Support

Silicon Graphics provides a comprehensive product support and maintenance program for IRIS series 3000 products. For further information, contact Customer Service through the Geometry Hotline

Silicon Graphics Geometry Hotline	
(800) 252-0222	U.S. except California
(800) 345-0222	California
(800) 443-0222	Canada

1.4 Conventions

This document uses the standard UNIX convention when referring to entries in the UNIX documentation. The entry name is followed by a section number in parentheses. For example, *cc*(1) refers to the *cc* manual entry in Section 1 of the *UNIX Programmer's Manual, Volume IA*.

In command syntax descriptions and examples, square brackets surrounding an argument indicate that the argument is optional. Variable parameters are in *italics*. You replace these variables with the appropriate string or value.

In text descriptions, filenames and UNIX commands are also in *italics*. IRIS Graphics Library routines and PROM commands are in `typewriter font`.

2. Hardware Installation

This chapter describes how to install and connect the standard components of the IRIS 3000 series systems.

2.1 Hardware Components

A typical configuration for IRIS series 3000 systems includes these components:

- Electronics cabinet with one disk drive
- Nineteen-inch tilt-and-swivel 60-Hz non-interlaced monitor (standard) or fifteen-inch tilt-and-swivel 60-Hz non-interlaced monitor
- Keyboard with right-hand and left-hand connectors for the mouse
- Optical mouse with grid pad
- Ethernet transceiver

For a more detailed description of the components, see Appendix A.

Options for the IRIS series 3000 products include:

- Dial and button box
- 33-Hz interlaced monitor
- Digitizer tablet
- Tektronix 4692 color printer
- Half-inch tape drive
- Quarter-inch cartridge tape drive

- Stereo optic viewer cable
- Floppy disk drive (standard on the IRIS 3010)
- Additional hard disk drive
- Light pen

You can use this manual for the IRIS 2300T, 2400T, and 2500T. Early IRIS series 2000 models with the Turbo upgrade may not have the same standard peripherals described here. The 60-Hz monitor may be different from the one described in this guide. The mouse may be mechanical rather than optical. The early IRIS 2400 Turbo system required a separate junction box. For a description of these features, see the *IRIS Workstation Guide Series 2000, Version 1.0*.

To install optional peripherals, see Chapter 7.

The major components have the following functions and features:

- The *electronics cabinet* for the standard configuration of the IRIS series 3000 is a floor-standing unit with a 20-slot backplane, a disk drive, and a power supply (see Figure 2-1). The cabinet uses forced-air cooling and is mounted on casters. A quarter-inch cartridge tape drive (internal) or a half-inch streaming tape drive (external) is optional on the IRIS workstation (see Figure 2-2). A floppy disk drive is standard on the IRIS terminal (see Figure 2-3). See Chapter 6 for disk specifications and Chapter 7 for tape specifications.

- Three types of *monitors* are available from Silicon Graphics for the IRIS series 3000. The fifteen-inch and nineteen-inch tilt-and-swivel monitors are 60-Hz high-resolution RGB color monitors. Silicon Graphics also offers a 33.43-Hz nineteen-inch interlaced monitor. (This monitor will be referred to as 33-Hz rather than 33.43-Hz.)

  You can also order the IRIS series 3000 with drivers for other types of monitors (see Chapter 8, Video Options).

- The *keyboard* is an 83-key, up-down encoded keyboard, with right-hand and left-hand connectors for the optical mouse.
- The *mouse* is a three-button optical mouse with a grid pad.

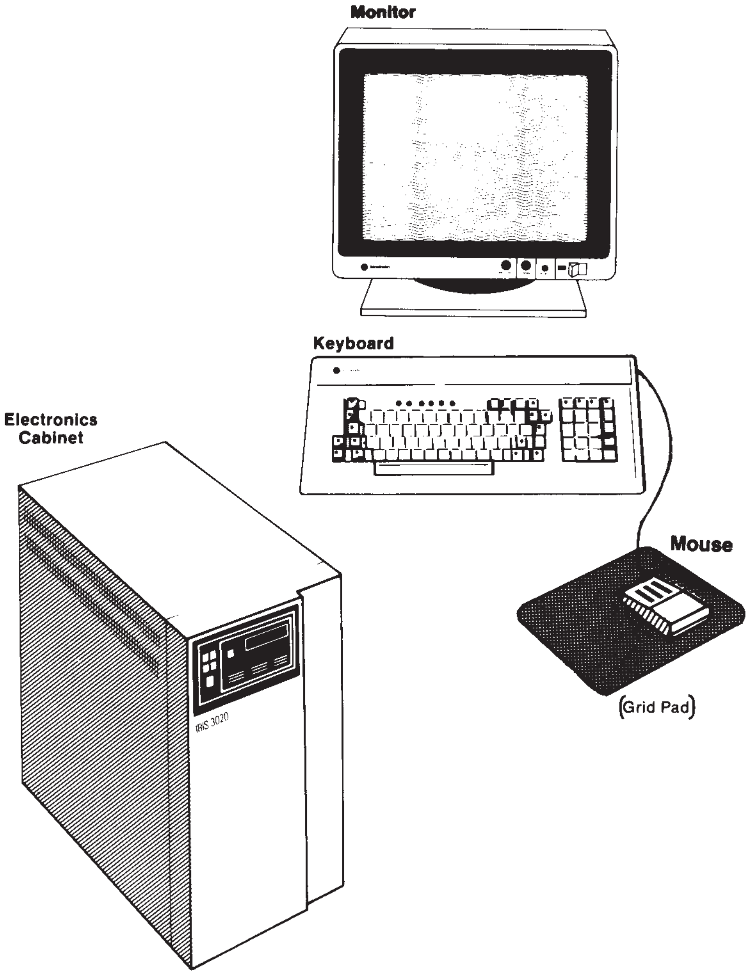


Figure 2-1: IRIS Series 3000 System

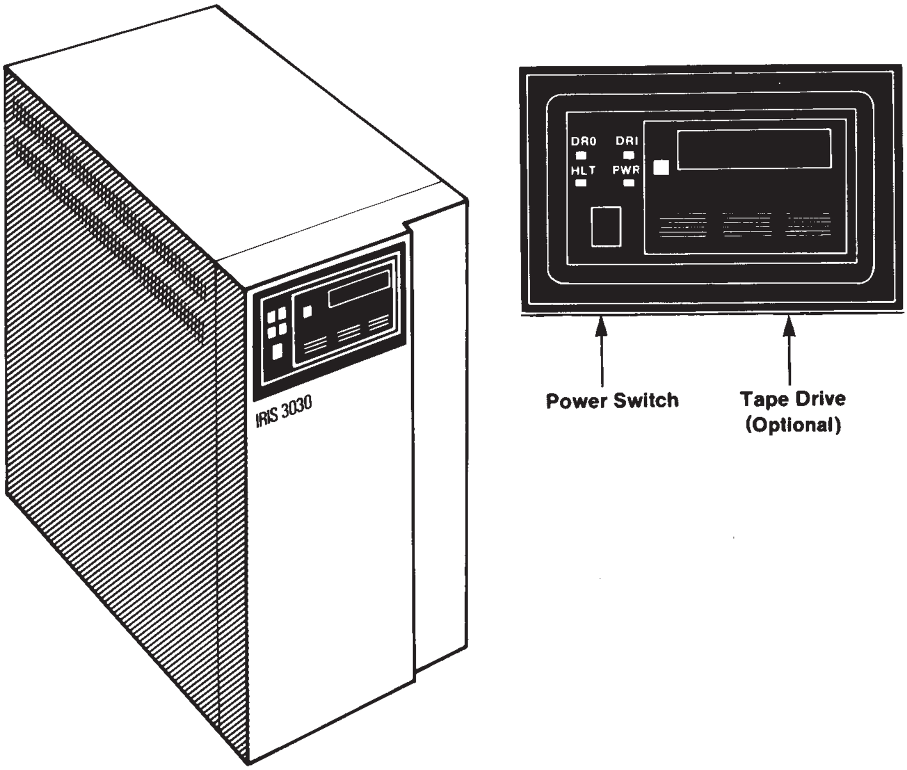


Figure 2-2: IRIS 3030 Cabinet Front Panel

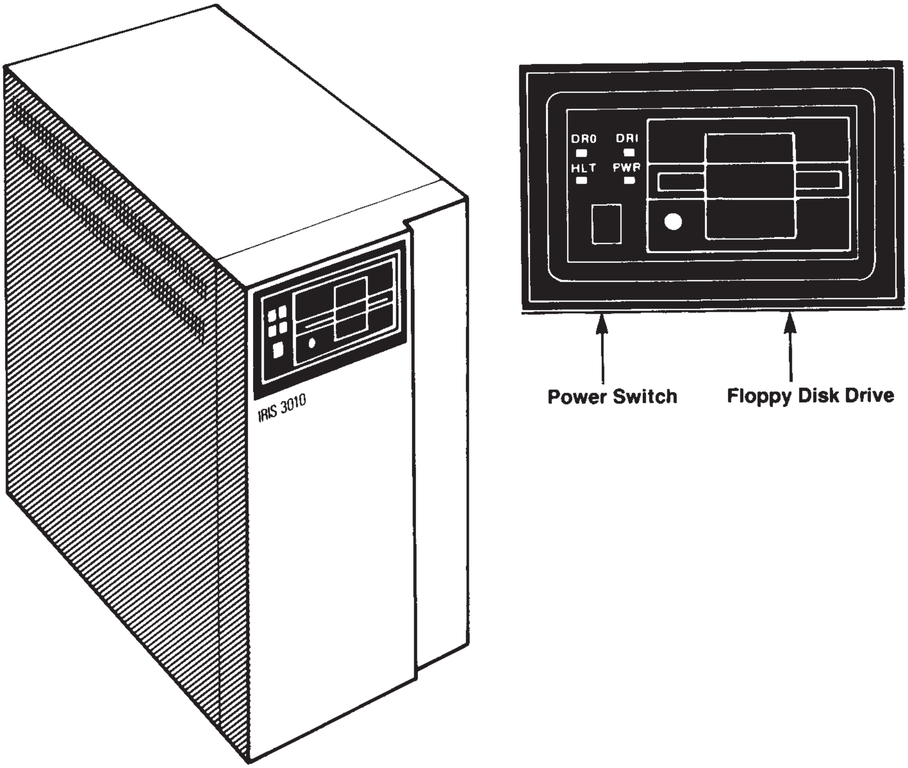


Figure 2-3: IRIS 3010 Cabinet Front Panel

2.2 Keyboard and Mouse Connection

You connect the mouse to the keyboard by inserting the mouse cable in the nine-pin D connector on either side of the keyboard. Secure the mouse connector to the keyboard with its two retaining screws. Then connect the keyboard to the IRIS by inserting the keyboard cable into `Port 1` of the standard I/O panel on the IRIS.

2.3 Monitor to Cabinet Video Connections

The procedure for connecting the monitor to the IRIS is different for each type of monitor. The sections below describe the procedures for each type of monitor.

2.3.1 Nineteen-inch 60-Hz Non-interlaced Monitor

Connect the monitor to the cabinet using the color-coded coaxial video cables (see Figure 2-4).

1. If you are using only one monitor, plug the terminators into the *unused* video connectors. The terminators are attached to the monitor by chains.

   To connect several monitors in a series (daisy chain), remove the terminators from all but the last monitor. Plug the terminators on the last monitor into the unused video input sockets.

   Attach the outgoing video cables to the unused BNC connectors on the back of the monitor.

2. Connect each color-coded cable end to the corresponding input socket on the monitor back panel.

   Push each cable into its connector and rotate to lock into place.

3. Connect the other end of each color-coded cable to the corresponding output socket on the standard I/O panel.

   Push each cable into the connector and rotate to lock into place.

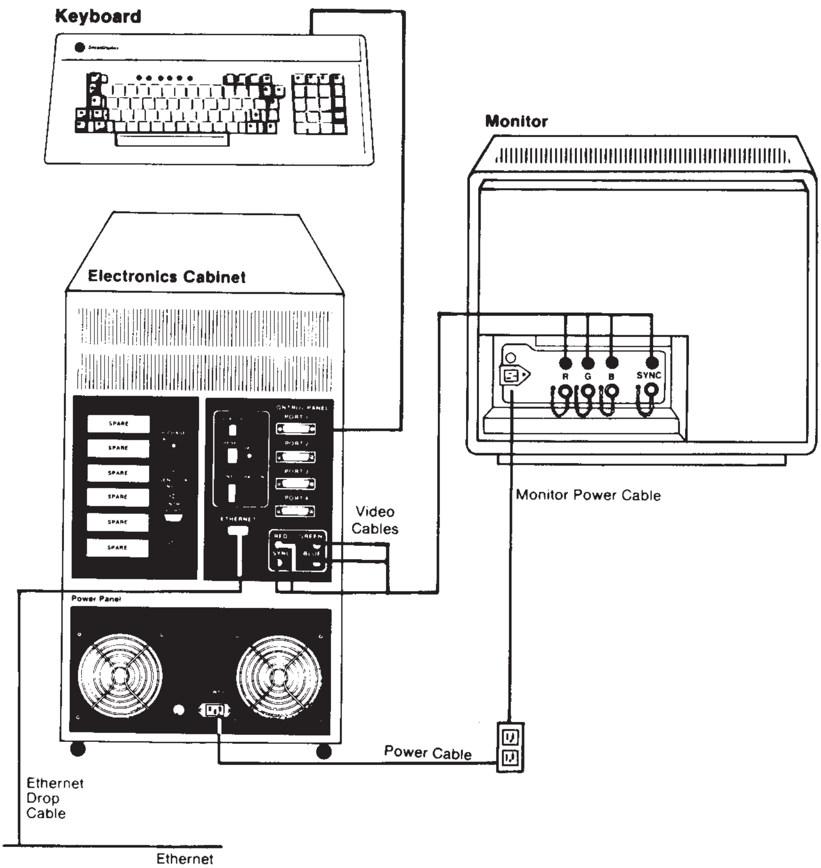


Figure 2-4: IRIS Cabinet Connections Nineteen-inch 60-Hz Monitor

2.3.2 Fifteen-inch 60-Hz Non-interlaced Monitor

Connect the monitor to the cabinet using the color-coded coaxial video cables (see Figure 2-5).

1. If you are using only one monitor, set *all* of the input impedance switches on the back of the monitor to the `75 Ω` position. The input impedance switches are push-button switches located between the input sockets. `IN` is the `75 Ω` position. `OUT` is the `High` position.

   To connect several monitors in a series (daisy chain), set the input impedance switches to the `High` position for all but the last monitor, which should be set to the `75 Ω` position.

   Attach the outgoing video cables to the unused BNC connectors on the back of the monitor.

2. Connect each color-coded cable end to the corresponding input socket on the monitor back panel.

   Push each cable into its connector and rotate to lock into place.

3. Connect the other end of each color-coded cable to the corresponding output socket on the standard I/O panel.

   Push each cable into the connector and rotate to lock into place.

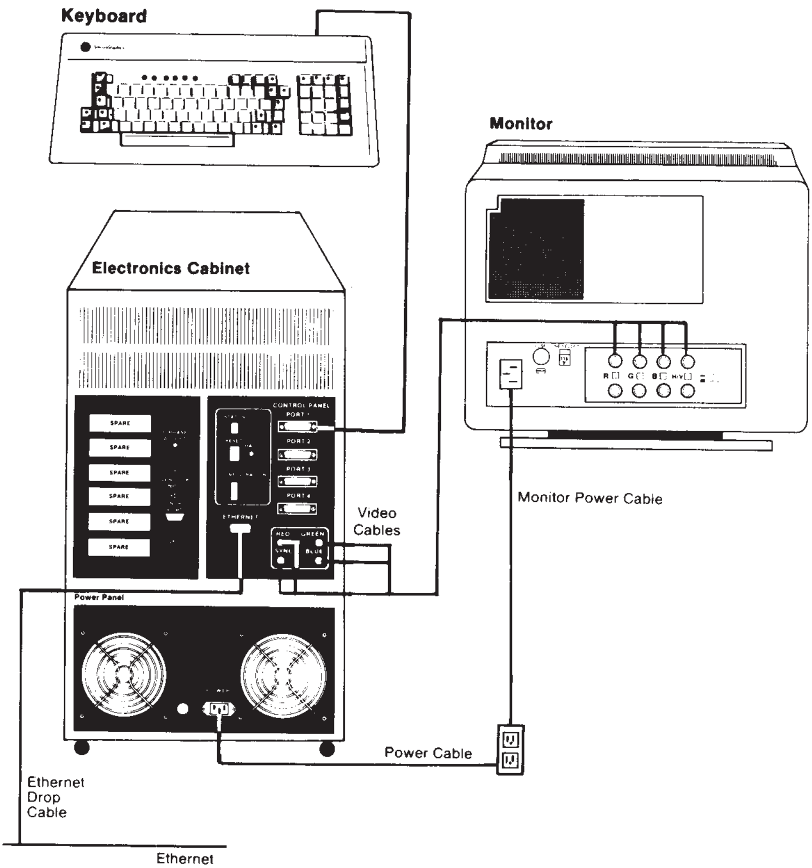


Figure 2-5: IRIS Cabinet Connections Fifteen-inch 60-Hz Monitor

2.3.3 33-Hz Interlaced Monitor

Use the color-coded coaxial video cables to connect the monitor to the cabinet (see Figure 2-6).

1. If you are using only one monitor, set *all* of the input impedance switches on the back of the monitor to the `75 Ω` position.

   To connect several monitors in a series (daisy chain), set the input impedance switches to the `High` position for all but the last monitor, which should be set to the `75 Ω` position.

   Attach the outgoing video cables to the unused BNC connectors on the back of the monitor.

2. Connect each color-coded cable end to either one of the corresponding input sockets on the monitor back panel.

   Push each cable into its connector and rotate to lock into place.

3. Connect the other end of each color-coded cable to the corresponding output socket on the standard I/O panel.

   Push the cable into the connector and rotate to lock into place.

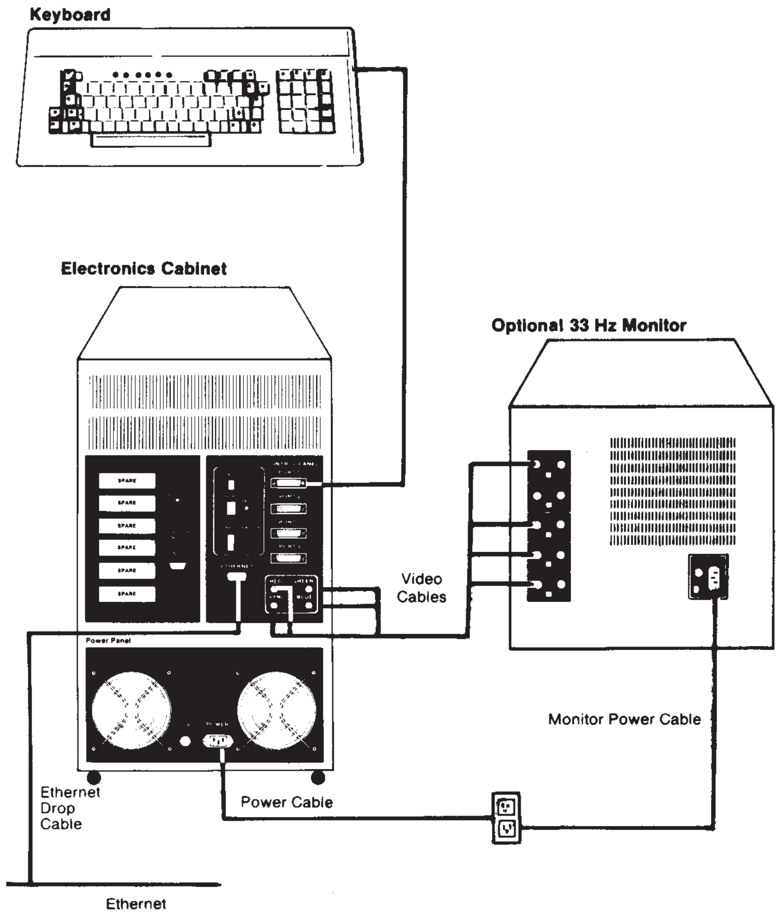


Figure 2-6: IRIS Cabinet Connections 33-Hz Interlaced Monitor

2.4 Monitor Adjustment

The monitor control panel on the right side of the monitor allows the characteristics of the monitor to be adjusted.

The nineteen-inch monitors (60-Hz and 33-Hz) have two knobs labeled `Brightness` and `Contrast`, and a button labeled `Degauss` (see Figures 2-7 and 2-8).

The fifteen-inch monitor has two recessed dials labeled `Brightness` and `Contrast`, and a button labeled `Degauss` (see Figure 2-9).

NOTE: Color rendering and stability may drift for the first 45 minutes after powerup.

1. After the monitor has warmed up, turn the `Brightness` and `Contrast` controls to the maximum (clockwise) setting.

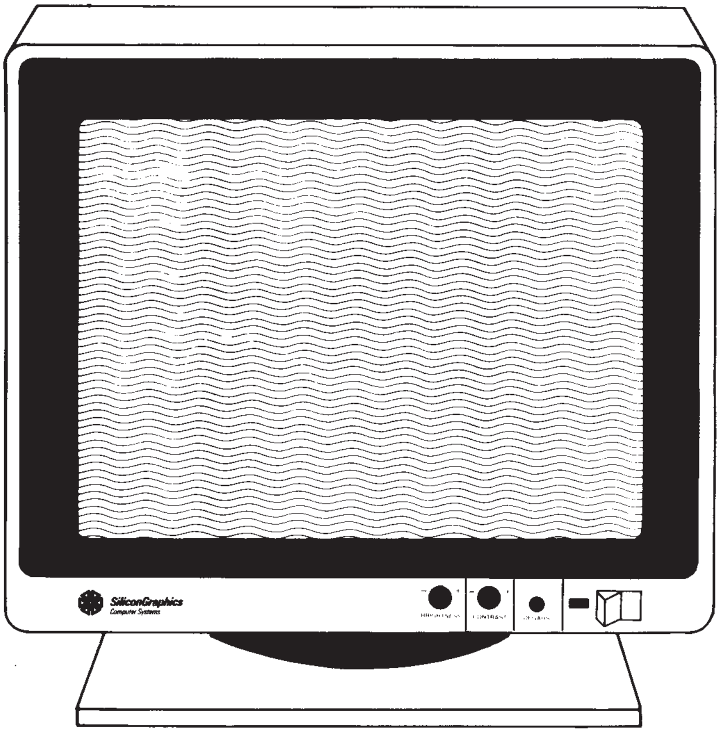
2. Turn the `Brightness` control down until the gray raster just disappears in relation to the black area at the edge of the screen. Lighter brightness settings impair image sharpness and color fidelity.

3. After the `Brightness` control has been set, adjustment of the `Contrast` control is a matter of personal taste.

4. If the color purity or convergence appear out of adjustment, hold down the `Degauss` button on the monitor control panel for about five seconds and then release it. The image should noticeably improve.

   Once you adjust the nineteen-inch 60-Hz monitor, push in the `Brightness` and `Contrast` knobs so that they are flush with the surface of the monitor.

Optional 19" monitor



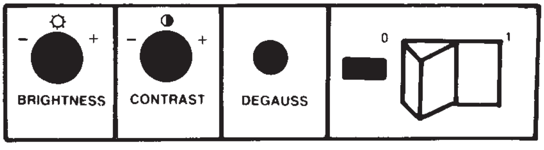


Figure 2-7: Control Panel for Nineteen-inch 60-Hz Non-interlaced Monitor

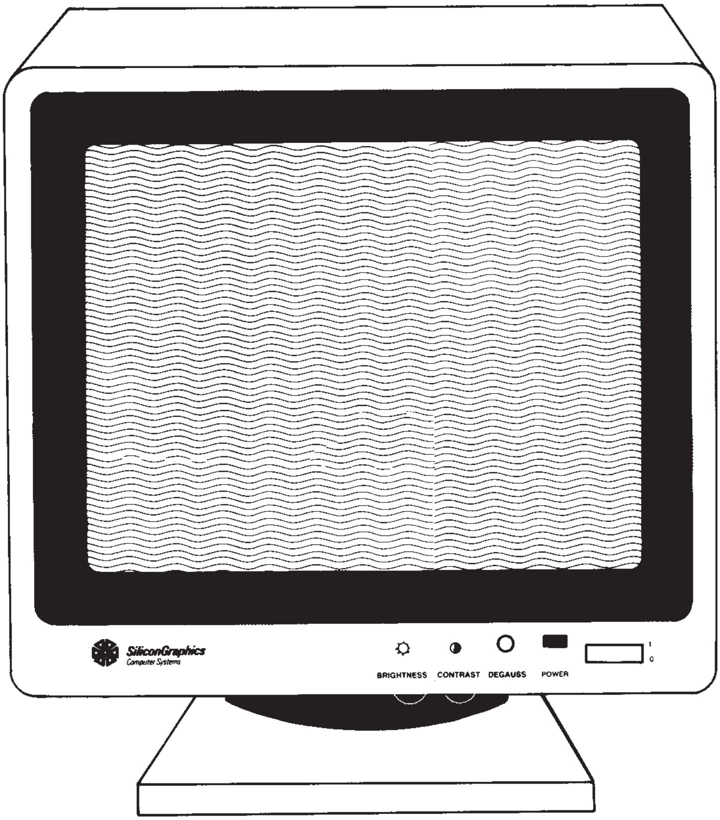


Figure 2-8: Control Panel for Fifteen-inch 60-Hz Non-interlaced Monitor

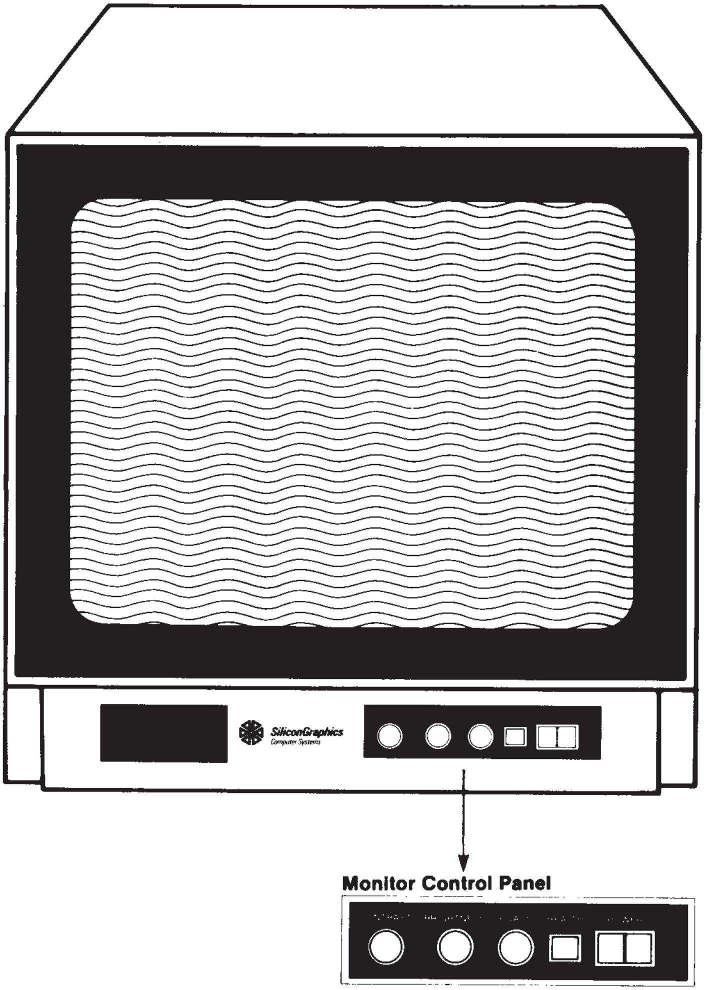


Figure 2-9: Control Panel for 33-Hz Interlaced Monitor

2.5 Monitor AC Power Cable Connection

Plug the monitor into a 120/240 VAC outlet.

1. Connect the female end of the AC power cable to the `Input` power socket on the monitor back panel.
2. Connect the male end of the monitor power cable to a wall outlet.

2.6 Cabinet AC Power Connection

The cabinet power socket is located on the cabinet power panel (see Figure A-4).

CAUTION: Do not connect the IRIS to a switched power outlet.

1. Connect the female end of the AC power cable to the power socket labeled `Power` on the cabinet power panel.
2. Connect the male end of the cabinet power cable to an appropriate outlet. See Appendix A, IRIS Specifications, for the IRIS power requirements.

2.7 IRIS to Ethernet Connection

The IRIS can communicate with other hosts and terminals (which have the necessary communications software) through an Ethernet local area network. You can connect the IRIS to an Ethernet local area network while the network is operating.

To connect the IRIS to an Ethernet local area network, you need:

- An *Ethernet transceiver* to attach to the Ethernet.
- One seventy-five foot, fifteen-conductor drop cable to connect the IRIS cabinet to the Ethernet transceiver.

The Ethernet port is located on the standard I/O panel, on the back of the cabinet.

Connect the Ethernet cable to the cabinet:

1. Connect the male end of the drop cable to the Ethernet port, labeled `Ethernet`, on the standard I/O panel. Use the slide lock to secure the cable in the connector.

2. Select an appropriate tap point on the Ethernet coaxial cable (instructions are included with each transciever).

NOTE: Approved Ethernet coaxial cable is marked with rings at 2.5-meter intervals (minimum distance). Transceivers should be placed at these rings to minimize transceiver reflection, which can induce transmission errors.

3. Connect the transceiver to the Ethernet cable.

4. Connect the female end of the drop cable to the transceiver.

3. Booting the IRIS

This chapter contains step-by-step procedures for booting and checking out a new IRIS system. This chapter also describes the bootstrap options and the startup environment for the IRIS. Before following the procedures in this chapter, you must install your system as described in Chapter 2. After you complete the procedures in this chapter, see Chapter 4 for workstation system administration procedures, or Chapter 5 for terminal system administration procedures.

This chapter contains the following information:

- Workstation Booting Instructions
- Terminal Booting Instructions
- The Startup Environment
- Boot Options

3.1 Workstation Booting Instructions

To boot the default software, which is on the disk, and run the flight simulator demonstration program, follow these steps:

1. Set the `Power` switch on the display monitor to `On`.
2. Set configuration switches 1 through 9 on the cabinet back panel to `Closed` (see Figure 3-1).

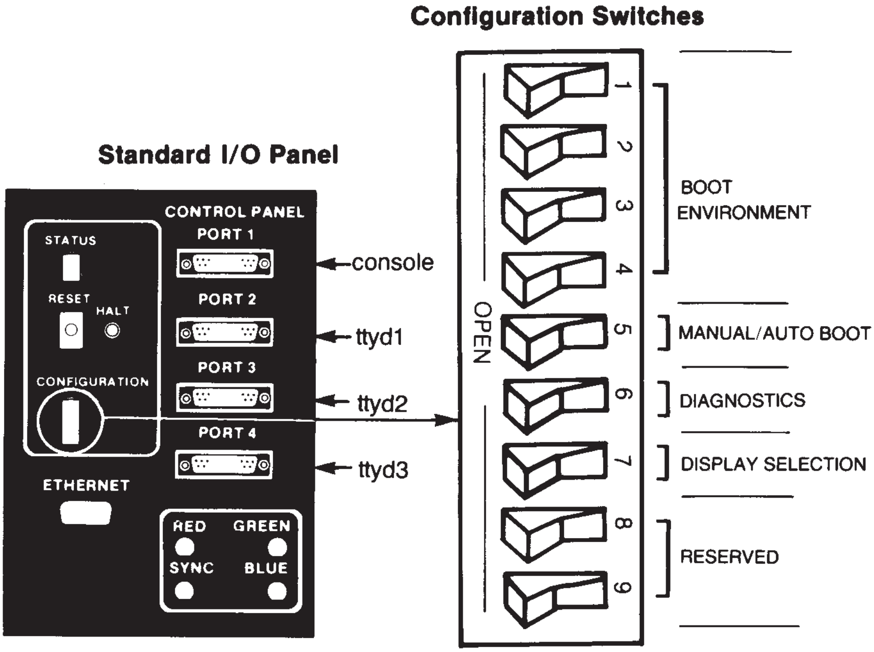


Figure 3-1: IRIS Configuration Switches

3. Set the `Power` switch on the cabinet front panel to `On`. If the power for the workstation is already on, press the `Reset` button. The workstation displays the PROM monitor prompt:

```
iris>
```

Wait 20 seconds tor the disk to spin up to speed. (The boot will fail if you don't wait that long.)

4. Enter the letter `b` and press `Return`.

```
b
```

The workstation displays system information similar to that shown in Figure 3-2. The IRIS is running in UNIX *single-user mode*, which is used only for system maintenance.

At the # prompt, start *multi-user mode* by typing:

```
multi
```

5. When the workstation prompts for the correct date, enter the date in the requested format, as illustrated in the example below. The characters *mmddhhmmyy.ss* represent digits for month-day-hour-minute-year-seconds. The year and seconds are optional. If the date shown is incorrect, type `n` and enter the correct date.

```
Is the date Wed July 21 08:10:33 PST 1989 correct? (y or n) n
Enter the correct date (mmddhhmm[yy][.ss]):0721083089.33
```

6. The workstation prompts for a file system check:

```
Do you want to check filesystem consistency? (y or n)
```

The system now performs initializations required for multi-user mode. Once multi-user mode has been started, a login prompt appears:

```
IRIS login:
```

8. To test a newly installed system, log in to the guest account:

```
guest
```

9. To use the Flight simulator, type:

```
~demos/flight
```

10. To leave the *flight* program, press ESC.

11. To log out, type:

```
logout
```

Important: Now that your IRIS workstation is up and running, make sure you have a bootable backup tape containing the stand-alone software, the root file system, and the user file system. You should receive a bootable backup tape with your workstation. This tape can be used to boot the workstation and rebuild the disk in case the file system is damaged.

For more information about system administration, such as adding new accounts, changing the time zone, and enabling network communication, see Chapter 4.

Section 3.4 contains more detailed information about the boot procedure, including procedures for booting from various devices. Section 3-4 also describes the PROM monitor, a command interpreter that controls the boot environment. Table 3-2 shows the commands understood by the PROM monitor.

Kernel Number:

```
SYSTEM 5 UNIX #6: [Fri May 4 11:15:09 PST 1985]
```

Copyright:

```
(c) Copyright 1983 - UniSoft Corporation
(c) Copyright 1983 - Silicon Graphics Inc.
```

Actual Memory:

```
real = 4194304
```

Kernel Size:

```
kmem = 491520
```

Available User Memory:

```
user = 3702784
bufs = 819200 (max=8k)
```

Hardware Configuration:

```
dsd0 at mbio 0x7f00 ipl 1
qic0 slave 0
md0 (Priam V170 Name: IP-2) slave 1
md1 (Priam V170 Name: IP-2) slave 2
mf0 not installed
st0 not installed
nx0 (FW2.6 HW 2.0) (0800.1400.0120) at mbio 0x7ffc ipl2
fpa installed
```

Root File System Device Name:

```
root on md0a
```

Swap Space Device Name and Size:

```
swap on md0b [8865K]
```

Figure 3-2: Sample IRIS Boot Information

3.2 Terminal Booting Instructions

To boot the default software on the disk and run the flight simulator demonstration program, follow these steps:

1. Set the `Power` switch on the display monitor to `On`.
2. Set configuration switches 1 through 9 on the cabinet back panel to `Closed` (see Figure 3-1).
3. Set the `Power` switch on the cabinet front panel to `On`. If the power is already on, press the `Reset` button. The IRIS terminal displays the PROM monitor prompt:

   ```
   iris>
   ```

   Wait 20 seconds for the disk to spin up to speed. (The boot will fail if you don't wait that long.)
4. Enter the letter `b` and press `Return`.

   ```
   b
   ```

5. When the terminal prompts for the correct date, enter the date in the requested format, as illustrated in the example below. The characters *mmddhhmmyy.ss* represent digits for month-day-hour-minute-year-seconds. Year and seconds are optional. If the date shown is incorrect, type `n` and enter the correct date.

```
Is the date Wed July 21 08:10:33 PST 1989 correct? (y or n) n
Enter the correct date (mmddhhmm[yy][.ss]):0721083089.33
```

6. After the startup sequence is complete, the IRIS 3010 automatically runs the terminal emulator program, *wsiris*. To enter the shell type in `~!` at the first prompt:

   TCP/IP:

   ```
   Enter IRIS IP address : ~!
   ```

XNS or IEEE-488 option:

```
Connect to what host? ~!
```

If your IRIS has the IBM option, type in `~!` to return from the *t3279* menu to the shell.

The shell prompt appears:

```
%
```

7. To use the flight simulator, type in:

```
~demos/flight
```

8. To leave the *flight* program, press ESC.

To return to the terminal emulator program, enter the `exit` command.

See Chapter 5 for more information on using the terminal emulator, connecting to a host, and configuring the user environment.

3.3 The Booting Environment

During the startup process, the IRIS displays a set of system information, including memory size, hardware configuration, and file system identification (see Figure 3-2).

The IRIS terminal boots into the terminal emulator program. For more information on IRIS terminal operation, see Chapter 5.

The IRIS workstation boots into single-user mode, which you use only for system maintenance. For normal operation, the workstation must be in multi-user mode. The *multi* command initiates the process of putting the workstation into multi-user mode. The details of the procedure are controlled by the file */etc/rc*, which contains commands tor starting daemons and mounting file systems. For more information, see *brc*(1).

To put the IRIS in multi-user mode, follow the procedures below:

1. Enter the *multi* command:

   ```
   multi
   ```

   If the */etc/model* file is unusable, the system prompts for the model number, writing the response to */etc/model*.

2. A prompt asks to run the UNIX file system check program *fsck*.

   ```
   Do you want to check filesystem consistency? (y or n)
   ```

   Normally, you should answer `yes`, and the *fsck* program checks the file systems. Answer `no` only if the system has been shut down with the *reboot* command, as described in Section 4.1. Even then, you are advised to run *fsck* occasionally.

   If *fsck* finds no problems, it displays one line for each test. If *fsck* finds any problems, it prompts for permission to repair the file system. See the *UNIX Programmer's Manual* for more information on *fsck*.

3. When the terminal prompts for the correct date, enter the date in the requested format, illustrated in the example below:

```
Enter the correct date (mmddhhmm[.ss][yy]):07210830.3389
Is the date Wed July 21 08:30:33 PST 1989? (y or n) y
```

   The characters *mmddhhmm.ssyy* represent month-day-hour-minute-seconds-year. The seconds and the year are optional. The entry above sets the date to 33 seconds past 8:30 a.m. on July 21, 1989.

   The system now performs initializations required tor multi-user mode. Once multi-user mode has been started, a login prompt appears:

   ```
   IRIS login:
   ```

NOTE: To configure your workstation so you can automatically boot into multi-user mode, see Section 4.2

3.4 Boot Options

You define the IRIS boot environment by setting configuration switches 1 through 4 on the cabinet back panel. The two basic boot options are automatic boot or PROM monitor boot:

Automatic boot	At powerup, the IRIS tries to boot from a file called *defaultboot* on the device specified by the configuration switches (see Table 3-1 and Section 3.4.2).
PROM monitor boot	At powerup, the IRIS enters the PROM monitor and waits for further boot instructions (see Section 3.4.1).

To boot from a secondary video driver, see Chapter 8.

Since the IRIS can be booted from different devices (hard disks, tape drives, etc.), the PROM monitor provides the `ls` command, a version of the UNIX *ls*(1) command, for displaying the names of the files on the attached devices. You can specify a device name and a pathname in the `ls` command (see Table 3-2).

For example, `ls /` searches the root file system (/) on the default device and lists its contents. If you type `ls` without an argument, the IRIS displays a list of files on the default device.

The IRIS is shipped with the configuration switches set for booting from the PROM monitor. To select automatic booting, change the settings of the configuration switches on the standard I/O panel. Table 3-1 lists the defined configuration switch settings.

Configuration Switches			
Switch	*Switch Name*	*Position*[1]	*Meaning*
1 - 4	Boot environment	CCCC OCCC COCC OOCC OCOC	Hard disk boot Cartridge tape boot Floppy disk boot Network boot PROM monitor
5	Autoboot	C O	PROM Monitor boot Automatic boot
6	Quiet mode	C O	Display system information Don't display system information
7	Monitor select	C O	Display on primary monitor Display on secondary monitor
8-9	Reserved	C	

Table 3-1: IRIS Configuration Switches

If you attempt to boot a nonexistent device or file from the PROM monitor, the system prints an error message and returns you to the PROM monitor. If an automatic boot fails, the system waits a few seconds and tries again. Automatically booting an IRIS 3030 always fails on the first attempt after power-up, but the automatic retry should succeed.

As Table 3-1 shows, switches 1 through 4 select the device from which the IRIS is to be booted. Switch 5 specifies whether the IRIS should perform an automatic boot or a PROM monitor boot. Switch 6 determines whether or not system information is displayed on the screen after the IRIS is reset.

[1] C means Closed and O means Open.

Switch 7 selects the display monitor type. (For an explanation of selecting a display type, see Chapter 8, Video Options.)

Device	Description
hd0	Disk drive 0 (generic)
md0	Disk drive 0 (IRIS 3010/3020)
si0	Disk drive 0 (IRIS 3030)
hd1	Disk drive 1 (generic)
md1	Disk drive 1 (IRIS 3010/3020)
si1	Disk drive 1 (IRIS 3030)
ct0	Cartridge tape drive 0 (generic)
mq0	Cartridge tape drive (IRIS 3020/3030)
sq0	Cartridge tape drive (IRIS 3030)
fd0	Floppy disk 0 (generic)
mf0	Floppy disk 0 (IRIS 3020/3030)
sf0	Floppy disk 0 (IRIS 3030)

Table 3-2: Boot Devices

3.4.1 PROM Monitor Boot

The PROM monitor is a command interpreter designed primarily to control the IRIS boot environment. The IRIS enters the PROM monitor when the power is turned on or the system is reset in either of two cases:

- Configuration switches 1 through 4 are set for PROM monitor boot (OCOC).
- Switch 5 is set to C, regardless of the setting of switches 1 through 4.

Table 3-2 lists the device names recognized by the PROM monitor. Table 3-3 summarizes the PROM monitor commands.

To boot through the PROM monitor, follow these steps:

1. Set the `Power` switch on the display monitor to `On`.
2. Set configuration switches 1 through 4 to the appropriate boot device setting (see Table 3-1). The default boot device is the hard disk, which has the setting CCCC.
3. Set switch 5 to `Closed`.
4. Set the `Power` switch on the cabinet front panel to `On`.

   If the clock battery has run down, the system displays this message, which refers only to the clock:

   ```
   Power has been lost to the machine.
   ```

   The IRIS displays the PROM monitor prompt:

   ```
   iris>
   ```

   If the prompt does not appear, press the `Reset` button.

   Wait 20 seconds for the disk to spin up to speed. (The boot will fail if you don't wait that long.)

5. To boot from the device specified by configuration switches 1 through 4, enter the letter `b` and press `RETURN`.

   The IRIS reads the file *defaultboot* from the default device.

Command	Description
`b`	Boot *defaultboot* from the default boot device.
`[b]` *file*	Boot *file* on the default boot device, determined by the configuration switches (see Table 3-1).
`[b]` *dev*`:`	Boot file *defaultboot* from specified device.
`[b]` *dev*`:` *file*	Boot *file* on device *dev*.
`h`	Display a list of PROM monitor commands.
`ls [`*dev*`:] [`*pathname*`]`	List the contents of directory *pathname* on device *dev*. With no arguments, *ls* provides a list of files on the default device (see Table 3-3).
`xns[.`*hostname*`]:[`*file*`]`	Boot *file* over the network using XNS.
`exit`	Restart the PROM monitor.

Table 3-3: PROM Monitor Commands

You can boot the IRIS from any attached device through the PROM monitor, regardless of the settings of switches 1 through 4. See Table 3-3 for a listing of the PROM monitor commands.

The examples below show how to boot from various devices using the PROM monitor. Table 3-2 lists the available device names. In all of the examples, if you enter a device name but no filename, the IRIS looks for a file with the name *defaultboot* on the specified device.

If you enter only a device or filename, or both, with no specific PROM monitor command, the IRIS tries to boot from the specified device and file.

Booting from the Tape Drive

To boot from the tape drive, specify the tape drive device name and filename. For example, to boot the file *mdfex* on the tape drive, enter this command:

```
ct0:mdfex
```

To be able to boot from a tape, it must be in *cpio* format. See Section 4.4 for information on making a bootable backup tape.

Booting from the Disk

To boot from the disk drive, specify the disk device name and filename. For example, to boot the file *vmunix* on the **a** partition of drive number zero, `hd0`, enter this command:

```
hd0:vmunix
```

Booting over the Ethernet

If you are running the XNS communications option on your IRIS, you can boot the IRIS over an XNS Ethernet. Use this command syntax:

`xns[`*.hostname*`]:`*filename*

If the hostname is omitted, the first host to recognize *filename* responds.

For example, to boot the IRIS from the file */usr/local/boot/goboot* on the host *cruncher*, enter this command:

```
xns.cruncher:/usr/local/boot/goboot
```

3.4.2 Automatic Boot

When configuration switch 5 is set for automatic boot, the IRIS attempts to boot the file *defaultboot* on the device specified by configuration switches 1 through 4 (see Table 3-1).

To configure the IRIS to boot automatically:

1. Set the `Power` switch on the display monitor to `On`.
2. Set configuration switch 5 to `Open`.
3. Set configuration switches 1 through 4 to the appropriate settings for the boot device (see Table 3-1).
4. Set the `Power` switch on the cabinet front panel to `On`.

NOTE: It the power for the IRIS is already on, press the `Reset` button. Do not press the `Reset` button while the IRIS is running UNIX without first using the */etc/reboot* command.

4. Workstation System Administration

The system administrator is responsible for configuring the IRIS workstation to meet user requirements and for maintaining the installed system. This chapter covers the following procedures:

- Basic operation and shutdown
- UNIX configuration:
  - –Naming the workstation
  - –Adding user accounts
  - –Booting into multi-user mode automatically
  - –UNIX configuration files
  - –Special device files in */dev*
- Network communication:
  - –TCP/IP network commands
  - –XNS network commands
  - –Using multiple kernels
- Backups:
  - –Making a bootable tape
  - –Making periodic backups
- Recovery from a crash

4.1 Basic Operation and Shutdown

If you have not installed and booted your workstation, see Chapters 2 and 3. For a description of the startup environment, see Section 3.3.

The IRIS workstation boots into single-user mode, which you should use only for system maintenance. For normal operation, the workstation must be put in multi-user mode, as described in Section 3.3. Section 4.2 explains how to configure the workstation to boot into multi-user mode automatically.

CAUTION: Do not press the `Reset` button while the IRIS workstation is running UNIX. For information on rebooting the system, see Chapter 3. If the IRIS workstation is not running UNIX and is under control of the PROM monitor, then you can use the `Reset` button or the `Power` switch.

Before turning off power to the workstation, you should perform a soft reboot of the system to ensure that all in-memory disk-blocks have been written to the disk and that all directory information on the disk is current.

To shut down the workstation, follow these steps:

1. Become the super-user.

   ```
   su
   ```

2. Make sure there are no other users on the system:

   ```
   who
   ```

3. Save all of the information that is in the memory buffer:

   ```
   sync
   ```

4. Reboot the system:

   ```
   reboot
   ```

5. Set the `Power` switches on the cabinet and monitor to `Off`.

4.2 UNIX Configuration

One of the strengths of the UNIX operating system is its flexibility. The system administrator configures the workstation to the needs of the local user community, and each user can fine-tune their own environment.

The following subsections contain instructions for the following common system administration tasks on an IRIS workstation:

- Naming a workstation
- Adding a new account
- Making a home directory
- Defining user groups
- Creating a password
- Creating a user environment
- Entering multi-user mode automatically

To perform these tasks, you must be logged in as *root* or *superuser*. These are two different names for the same account. When you are root (superuser), you can access and change all fo the files and directories on the workstation. You can become root in two different ways:

- Log in as `root` at the `IRIS login` prompt, or
- Type `su` when you are logged in to your own account. When you want to return to your own account, type `exit`.

If you are not logged in, use the first method to become root. At the login prompt, type:

```
root
```

For more information on UNIX configuration, see the *UNIX Programmer's Manual* and the list of configuration files in Section 4.2.4.

4.2.1 Naming an IRIS Workstation

The default name of a new IRIS workstation is *IRIS*. If you have more than one workstation on a network, you must assign each workstation a unique name. The name can be up to eight characters long and must contain no blanks.

To change the name of a workstation, edit the file */etc/sys_id*. Replace the word *IRIS* with the new name for your system. Save the file, then reboot the system. To display the system name at login, edit the file */etc/gettydefs*. Replace all occurrences of *IRIS* with your new system name. To display a message at login, edit the file */etc/motd* (message of the day). The IRIS will display both the new login prompt and the new message the next time you log in. To determine the current name of the system, use the *hostname*(1) command.

4.2.2 Adding a New Account

The IRIS workstation is shipped with seven user accounts: *root*, *rootcsh*, *rootsh*, *guest*, *demos*, *mexdemos,* and *tutor*. The *root* and *rootcsh* accounts are super-user accounts with a C shell environment. The *rootsh* account is a super-user account with a Bourne shell environment. The *guest* account is a sample user account with a C shell environment. The *demos* and *mexdemos* accounts are for running demonstrations (see Chapter 9). The *tutor* account contains the software that accompanies the *IRIS Programming Tutorial, C Edition*.

The accounts shipped on a new IRIS workstation have no passwords. To establish passwords tor these accounts, follow the procedure below. See the *passwd*(1) manual page for more information.

If you have the Network File System option and want to set up the Yellow Pages accounts, see Chapter 3 of the *NFS User's Guide*.

Typically, each user on an IRIS workstation is given his or her own account. The procedure for creating a new account on the IRIS workstation consists of adding the new user to the file */etc/passwd*, creating a home directory tor the user, and setting up startup files to define the user's environment. The details of this procedure are presented below. You should become the superuser to create a new account (see above).

Creating New User Accounts

Add a line for the new account to the file */etc/passwd*. Figure 4-1 contains a sample */etc/passwd* file.

```
root::0:0:Superuser:/:/bin/csh
rootcsh::0:0:Superuser:/:/bin/csh
rootsh::0:0:Superuser:/:/bin/sh
daemon:*:1:1::/:
bin:*:2:2:Binary Files:/:
adm:*:5:3:Administration:/usr/adm:
iris::7:0::/:/bin/tesh
uucpadm:*:8:8:UUCP Administration:/usr/lib/uucp:
lp:*:9:9:Line Printer:/:
mexdemos::996:997::/usr/people/mexdemos:/bin/csh
demos::997:997::/usr/people/demos:/bin/csh
guest::998:998::/usr/people/guest:/bin/csh
games:*:999:999:Games:/usr/games:/bin/sh
steve::11:20:Steve Brown:/usr/people/steve:/bin/csh
```

Figure 4-1: Sample */etc/passwd* File

The file */etc/passwd* contains a line for each account on a UNIX system. Each line has seven fields separated by colons (:).

Field	Contents
1	Account name
2	Encrypted user password
3	User number
4	Group number
5	User's real name
6	Home directory (default /)
7	Login shell (default */bin/sh*)

Table 4-1: Fields in the */etc/passwd* File

Be sure that the new entry contains a unique account name, a unique user number, and a group number. You, as system administrator, decide on a new, unique user number for each user to assign ownership of files to each user.

The group number is the group identification given to the user at login. When the user logs in, he or she can access any files shared by all users in that group.

The group number corresponds to the groups in the file */etc/group* (see *group*(4)). The */etc/group* file shipped on the workstation has group number 20 set to *user*; this is a default group number to use for new users.

Leave the password field empty. The system encrypts the password and adds it to the file when the user establishes a password (see step 4). In Figure 4-1, the asterisks (*) in the password fields for some accounts prevent those names from being used as login accounts.

Ordinarily, the home directory in field 6 is the user's own directory (see step 2). The login shell in field 7 can be either the C shell (*/bin/csh*) or the Bourne shell (*/bin/sh*). The default is the Bourne shell.

For example, to add a new account for a user named *jim* to the sample file in Figure 4-1, you might enter this line:

```
jim::10:20:Jim Smith:/usr/people/jim:/bin/csh
```

See *passwd*(4) for more information.

Making a Home Directory

Make a home directory tor the new user, with the same name specified in field 6 of the */etc/passwd* entry, and set the protections and ownership of the new directory. Typically, all users' home directories are made subdirectories of the directory */usr/people*.

The *mkdir*(1) command creates the new directory.

```
    mkdir /usr/people/jim
```

The *chgrp*(1) command changes the group to which the directory belongs. The example below puts the directory */usr/people/jim* in the group *user*.

```
chgrp user /usr/people/jim
```

The *chmod*(1) command sets the protection parameters on a file or directory. The example below sets protection codes to allow only the owner to create files in the directory. All users can read the file and search the directory. Protection parameters can be set by either the super-user or by the owner of the file or directory.

```
chmod 755 /usr/people/jim
```

The *chown*(1) command changes the ownership of the directory. The ownership of a directory can be set either the superuser or the owner of the directory. The example below establishes *jim* as the owner of the directory */usr/people/jim*.

```
chown jim /usr/people/jim
```

Defining User Groups

To include the new user in groups other than *user*, edit the file */etc/group*. Entries to this file are optional. See *group*(4) for more information.

```
sys:*:0:
system:*:0:
daemon:*:1:
bin:*:2:
adm:*:3:
sgi_use:*:4:
uucp:*:5:uucp
sgi_use:*:6:
sgi_use:*:7:
uucpadm:*:8:uucp
lp:*:9:
user:*:20:
vlsi:*:21:steve
```

Figure 4-2: Sample */etc/group* File

Figure 4-2 contains an example */etc/group* file. The file contains a line for each group on the system. Each line consists of four fields separated by colons (:).

Field	Contents
1	Group Name
2	Encrypted Group Password
3	Group Number
4	Group Members

Table 4-2: Fields in the */etc/group* File

To add a user to a group, append the user's name to the list of group members. Separate group members with commas. For example, to add *jim* to the group *vlsi*, append the name *jim* to the line for the group *vlsi*:

```
vlsi:*:21:steve,jim
```

Creating a Password

Use the *passwd* command to establish a password for an account. If you want to set up a password for your personal account, log out so you are no longer the superuser, then log back in as yourself. Then type:

```
passwd
```

The IRIS asks you to enter a password, and then to confirm the password. The IRIS encrypts it and adds it to the file */etc/passwd*.

If you forget your password, as superuser you can remove the encrypted password from the entry in */etc/passwd*.

You can also use the *passwd* command to change passwords. The *passwd* program queries for the old password before allowing you to set up a new one.

Creating a User Environment

Add startup files like *.cshrc*, *.login*, and *.profile* to the new user's home directory. These files control the details of the environment. You can copy the startup files from */usr/people/guest*, and modify them if necessary.

The choice of the C shell or the Bourne shell determines which files are necessary. If you choose the C shell, the user's home directory should contain the files *.login* and *.cshrc*. The C shell reads the *.cshrc* file in the user's home directory each time a shell is started. If the shell is a login shell, the C shell then reads the *.login* file in the user's home directory.

If you choose the Bourne shell, the user's home directory should contain the file *.profile*. The Bourne shell reads the *.profile* file in the user's home directory when a login shell is started. For more information on the C shell and the Bourne shell, see *csh*(1) and *sh*(1).

Each user's shell startup file should use the *tset* command to read */etc/ttytype* and set the terminal type, so that screen editors and other programs know how to communicate with the terminal.

The usual way to use *tset* with the C shell is to add these lines to your *.login* file:

```
set noglob
set tmp = (`tset -S -Q`)
setenv TERM = tmp[1]
setenv TERMCAP = tmp[2]
unset tmp noglob
```

Add this line to your *.profile* file to use *tset* with the Bourne shell:

```
eval `tset -s -Q`
```

For more information, see *tset*(1) and *ttytype*(4).

4.2.3 Entering Multi-user Mode Automatically

By default, the IRIS workstation boots into single-user mode, which the superuser should use only for system maintenance. When the system is running in single-user mode, only the root file system is mounted, none of the system daemons are started, and the terminal lines do not allow users to log in. To enter multi-user mode, issue the *multi* command as the superuser.

As superuser you can also edit the file */etc/inittab* so that the system automatically enters multi-user mode each time it is booted. The first line of */etc/inittab* tells *init* which mode to enter after booting UNIX:

```
is:s:initdefault:
```

Replace the `s` (for single) in the second field with `2` or `3`:

TCP/IP protocol:

```
is:3:initdefault:
```

XNS protocol:

```
is:2:initdefault:
```

The system will now enter multi-user mode each time it is rebooted. For more information on the *init* process, see *init*(1M) and *inittab*(4).

4.2.4 UNIX Configuration Files

The directories */etc* and */usr/lib* contain the major system configuration files. They can be edited by the system administrator (superuser) as needed to serve the needs of the local user community. If you have the NFS option, see the *NFS User's Guide* for information about the configuration files that are affected by NFS or Yellow Pages.

/etc/bcheckrc — This file contains commands to perform a file system check and set the date using */etc/rc.getdate*. If it can't find */etc/rc.getdate*, it prompts for the date.

/etc/brc — The *brc* procedure performs model-specific system initialization chores.

/etc/cshrc — The system reads this file at login for accounts that specify the C shell as the login shell. It contains commands that define the user's environment. See *csh*(1) and *cshrc*(4).

/etc/ethers — This file contains information regarding the known (48-bit) Ethernet addresses of hosts on the internet.

/etc/exports — This file describes the file systems that are exported *nfs* clients.

/etc/fstab — The file */etc/fstab* describes the filesystems and swapping partitions used by the local IRIS. The system administrator can change *fstab* with a text editor. *fstab* is read by commands that mount, unmount, dump, restore, and check the consistency of filesystems. *fstab* encompasses the files */etc/checklist* and */etc/rc.fs* used in previous IRIS software releases.

/etc/gettydefs — This file contains entries for line speeds and terminal settings used by *getty*(1M) for initializing devices. See *gettydefs*(4).

/etc/group This file contains the list of groups that are used to control access to files. See Section 4.2 and *group*(4).

/etc/hosts This file contains information on the known hosts on the DARPA Internet.

/etc/hosts.equiv This file contains a list of trusted hosts. When an *rlogin* or an *rsh* request from such a host is made, and the intiator of the request is in */etc/passwd*, no further validity checking is done.

/etc/ibtab This file is an unordered collection of entries, each of which describes one IEEE 488 bus node. Its format is understood by the *ibtab*(3N) subroutine package, which is used by programs such as *iib*(1M).

/etc/inittab This file contains the table that *init* uses to dispatch processes for initializing devices, starting daemons, and starting the shell. See *inittab*(4), and *init*(1M).

/etc/model This file contains the model number of the IRIS.

/etc/motd This file contains the message of the day, which is displayed each time a user logs in to an IRIS workstation.

/etc/networks This file contains information regarding the known networks that constitute the DARPA Internet.

/etc/passwd This file contains information about the users with accounts on the workstation. It includes each user's name, user number, group number, home directory, and login shell. See Section 4.2.2 and *passwd*(1).

/etc/profile	This file is read at login by accounts that specify the Bourne shell as the login shell. It contains commands that define the user's environment. See *sh*(1) or *profile*(4).
/etc/protocols	This file contains information regarding the known protocols used in the DARPA Internet.
/etc/rc	This command file is executed by *init*(1M) at the start of multi-user mode. Typically, it is used to start daemons and mount file systems. See *brc*(1M).
/etc/rc.local	This file is for customized machine intialization procedures It is for local startup commands or daemons. This file does not change with each software update so that customized initialization procedures are not lost with each new software release. */etc/rc* runs */etc/rc.local* every time the IRIS is put into multi-user mode.
/etc/rc.getdate	If a file with this name exists and is executable, the standard output of its execution is used as the argument to *date*(1). Otherwise, the user is prompted for the date.
/etc/rpc	This file contains user readable names that can be used in place of *rpc* program numbers.
/etc/services	This file contains information regarding the known services available in the DARPA Internet.
/etc/sys_id	This file contains the name of the system, which is always *IRIS* on a new workstation. See Section 4.2, *hostname*(1), and *sys_id*(4).

/etc/termcap	This file contains definitions of different terminal types. See *termcap*(4).
/etc/syslogd	This file reads and logs messages into a set of files described by the configuration file */etc/syslog.conf*.
/etc/ttytype	This file contains a mapping of the terminal types attached to each tty port on the IRIS workstation. This file is read by *tset*(1) and by *login* to initialize the terminal (TERM) environment variable at login. See *tset*(1) and *ttytype*(4).

/etc/TZ	This file contains an entry for the time zone. Several different utilities use this file. */etc/TZ* contains three fields:
	
	Standard heading for time zone
	Offset from Greenwich Mean Time (in hours)
	Optional daylight savings time zone
	
	The IRIS workstation is shipped with the time zone set for Pacific Standard Time:
	
	`PST8PDT`
	
	These are examples for other time zones:
	
	`EST5EDT`
	`CST6CDT`
	`MST7MDT`
	
	For more information, see *TZ*(4).

/usr/lib/acct/holidays	This file contains a list of holidays. It is used by the *calendar*(1) program.
/usr/lib/crontab	This file contains entries for commands to be run at fixed intervals by the *cron* daemon (the clock daemon). See *cron*(1M).

/usr/lib/Mail.rc

This file contains *set* commands used for initializing the mail system.

The next four files configure the UNIX-to-UNIX communication system UUCP. Further information on UUCP can be found in the *UNIX Programmer's Manual, Volume II*.

/usr/lib/uucp/L-devices

This file sets the line speeds for the ports used by *uucp*(1C). The file contains a series of one-line entries that follow the format of this example:

```
DIR tty02 0 4800
```

This entry specifies that the device */dev/tty02* is a directly connected computer that can be used at 4800 baud.

/usr/lib/uucp/L-dialcodes

This file contains the dial-code abbreviations used in the */usr/lib/uucp/L.sys* file. The entries are in the form:

abb dial-sequence

abb is the abbreviation of a location, and *dial-sequence* is the dial sequence associated with the location. For example:

```
sf 415
```

This entry sends the sequence 415 to the dial unit.

/usr/lib/uucp/L.sys

This file contains information about sites that *uucp*(1) can communicate with.

/usr/lib/uucp/USERFILE This file contains information that limits user accessibility. It specifies which files a user on the local machine can access, which files can be accessed from a remote computer, which login name is used by a particular remote computer, and whether a remote computer should be called back to confirm its identity.

4.2.5 Device Files

The directory */dev* contains all of the device files. You never need to edit these files. Table 4-3 describes the contents of each file.

File	Description
drum	Paging device (used for virtual memory)
floppy	Optional floppy disk drive (generic blocked device)
rfloppy	Optional floppy disk drive (generic raw device)
ib[0-9]	IEEE 488 devices
kmem	Kernel memory; see *mem*(7)
md0[a-h]	Disk zero partitions (IRIS 3010/3020 block devices) **md0a** contains the root (/) file system; **md0c** contains the user (*/usr*) file system (IRIS 3020) or a copy of the root file system (IRIS 3010). See *efs*(4).
rmd0[a-h]	Disk zero partitions (IRIS 3010/3020 raw devices)
md1[a-h]	Optional disk partitions (IRIS 3010/3020 block devices)
rmd1[a-h]	Optional disk partitions (IRIS 3010/3020 raw devices)
mem	Memory; see *mem*(7)
mf0a	Optional floppy disk drive (IRIS 3010/3020 blocked device)
rmf0a	Optional floppy disk drive (IRIS 3010/3020 raw device)
null	Null device (zero length on input, data sink on output) See *null*(7).
pxd	IBM driver.

Table 4-3: Special Device Files in */dev*

File	Description
rmt1	Cartridge 1/4-inch magnetic tape (generic raw blocked device, rewinds at close)
rmt2	Cartridge 1/4-inch magnetic tape (generic raw blocked device and no rewind on open or close)
rmt3	Half-inch magnetic tape (raw blocked device, rewinds at close)
rmt4	Half-inch magnetic tape (raw blocked device with no rewind on open and close)
sf0a	Optional floppy disk drive (IRIS 3030 blocked device)
rsf0a	Optional floppy disk drive (IRIS 3030 raw device)
ip0[a-h]	Disk zero partitions (IRIS 2500T block devices)
ip1[a-h]	Optional disk partitions (IRIS 2500T block devices)
rip0[a-h]	Disk zero partitions (IRIS 2500T raw devices)
rip1[a-h]	Optional disk partitions (IRIS 2500T raw devices)
si0[a-h]	Disk zero partitions (IRIS 3030 block devices) **si0a** contains the root (/) file system; **si0c** contains the user (*/usr*) file system. See *efs*(4).
rsi0[a-h]	Disk zero partitions (IRIS 3030 raw devices)
si1[a-h]	Optional disk partitions (IRIS 3030 block devices)
rsi1[a-h]	Optional disk partitions (IRIS 3030 raw devices)
sq0	Cartridge magnetic tape (IRIS 3030 raw blocked device)
nrsq0	Cartridge magnetic tape (IRIS 3030 raw blocked device and no rewind on open or close)
syscon	System console (linked to */dev/console*)
systty	System console (linked to */dev/console*)
cent	Color graphics printer device.
vers	Versatec printer device
tek	Tektronix printer device

Table 4-3: Special Device Files in */dev* (continued)

File	Description
tty	A synonym for the terminal device associated with a process See *termio*(7) and *tty*(7).
ttyd1	Serial `Port 2` on standard I/O panel
ttyd2	Serial `Port 3` on standard I/O panel
ttyd3	Serial `Port 4` on standard I/O panel
ttym1	Serial `Port 2` for a modem
ttym2	Serial `Port 3` for a modem
ttym3	Serial `Port 4` for a modem
ttyf2	Serial `Port 3` flow control for a device that understands hardware flow control
ttyf3	Serial `Port 4` flow control for a device that understands hardware flow control
ttyn*	XNS Ethernet network terminal devices
ttyw[0-9]	Window ttys; *ttyw0* is the console.
ptc	Pseudo tty device
ttyq*	Pseudo tty devices

Table 4-3: Special Device Files in */dev* (continued)

4.3 Network Communication

An IRIS workstation can communicate with other devices over an Ethernet local area network using the TCP/IP protocol. You can purchase XNS protocol as an option. You can use only one communication protocol at one time.

If you have an IRIS 3020, 3030, 2400 Turbo, or 2500 Turbo configured for TCP/IP network communication, you can run the network file system option on your workstation. Sun Microsystem's Network Filesystem (NFS) allows file sharing in a heterogeneous environment of computers, operating systems, and networks. See the *NFS User's Guide* for more information if you have the NFS option on your IRIS.

The following sections present a brief overview of the commands you use to communicate over the Ethernet. For more information about the network software for the IRIS workstation, see the manual pages for these commands in the *UNIX Programmer's Manual, Volume IA*. To make the workstation emulate a terminal, see Chapter 5.

The following two sections use these terms:

- A *network* is a collection of computers and terminals connected together.
- A *host* is a computer on a network.
- A *local* host is the host you are using.
- A *remote* host is another computer on the same network.

4.3.1 TCP/IP Network Commands

This section describes the commands for communicating over an Ethernet local area network using TCP/IP. For a more complete description of how to use TCP/IP, see the *TCP/IP User's Guide*. You must be running the TCP/IP kernel to use TCP/IP communications (See Section 4.3.3).

Each computer to be accessed with the network communication tools must have a unique name. The file */etc/hosts* contains these names. (See the *NFS User's Guide* for more information.)

To communicate over an TCP/IP Ethernet local area network, the IRIS workstation uses five commands:

rcp	Copies a file from one computer running UNIX to another computer running UNIX.
rsh	Starts a shell on a remote host running UNIX.
rlogin	Initiates a login on a remote host running UNIX.

See the *TCP/IP User's Guide* for a complete list of *TCP/IP* commands.

rcp copies a file from one host to another. The command line specifies the source host and file, followed by the destination host and file:

```
rcp [sourcehost:]pathname [destinationhost:]pathname
```

If you don't specify a host name, network assumes you mean the local host. In the following examples, the user must have an account with the same user name on both hosts.

This example copies the file *sqiral.c* in the current directory of the local machine to the file *sqiral.c* in the directory */oh4/doc/install* on a host named *olympus*.

```
rcp sqiral.c olympus:/oh4/doc/install/sqiral.c
```

This example copies the file */usr/include/stdio.h* from the remote machine *sting* to the file *test* on the local machine.

```
rcp sting:/usr/include/stdio.h test
```

This example copies the file *temp_vi* from the remote machine named *puppy* to a file with the same name on a remote machine named *sting* in the user's home directory on each machine.

```
rcp puppy:temp_vi sting:temp_vi
```

This command is an example of recursive copying. It copies all files and directories from */usr/include* on the remote machine *sting* to the directory *localinclude* on the local host.

```
rcp -r sting:/usr/include/* localinclude
```

rsh connects your terminal to a remote host and executes the commands you specify. Like *rcp*, this network utility assumes that the user has accounts under the same user name on both the remote and local host.

If a command is specified, the *rsh* program takes the command as an argument. For example, to find the load average on another machine, enter this command:

```
rsh hostname uptime
```

rlogin initiates a login on a remote host across the network. The command takes the remote host name as an argument. For example, to log in remotely to a host named *olympus*, enter this command:

```
rlogin olympus
login:
...
```

If you do not specify a command to be executed, *rlogin* starts a login process on the remote host.

4.3.2 XNS Network Commands

For a more complete description of XNS network commands, see the *IRIS XNS User's Guide*, and the manual pages for these commands in the *UNIX Programmer's Manual, Volume IA*.

To communicate over an XNS Ethernet local area network, the IRIS workstation uses three commands:

xcp Copies a file from one computer to another.

xx Runs a command on a remote host.

xlogin Initiates a login on a remote host.

xcp copies a file from one host to another. The command line specifies both the source host and file first, followed by the destination host and file:

```
xcp [sourcehost:]pathname [destinationhost:]pathname
```

If you do not specify a host name, the network assumes you mean the local host. In these examples, the user must have an account with the same user name on both hosts.

The following example copies the file *sqiral.c* in the current directory of the local machine to the file *sqiral.c* in the directory */oh4/doc/install* on a host named *olympus*.

```
xcp sqiral.c olympus:/oh4/doc/install/sqiral.c
```

The following example copies the file */usr/include/stdio.h* from a remote machine *sting* to the file *test* on the local machine.

```
xcp sting:/usr/include/stdio.h test
```

The following example copies the file *temp_vi* from the remote machine named *puppy* to a file with the same name on a remote machine named *sting* in the user's home directory on each machine.

```
xcp puppy:temp_vi sting:temp_vi
```

The **–r** option invokes recursive copying, i.e. it copies all files in a directory. For example, it copies all files and directories from */usr/include* on the remote machine *sting* to the directory *localinclude* on the local host. The **–v** option displays the names of the files it creates. The asterisk (*) means "all files in the directory".

```
xcp -r -v sting:/usr/include/* localinclude
```

xx runs commands on a remote host. Like *xcp*, this command assumes that the user has accounts under the same user name on both the remote and local host. You can also log in as *guest* to any IRIS that has a guest account.

The *xx* command takes the command to be executed as an argument. For example, to find the load average on another machine, enter this command:

xx *hostname* uptime

xlogin initiates a login on a remote host across the network. The command takes the remote host name as an argument. For example, to log in remotely to a host with the name *olympus*, enter this command:

```
xlogin olympus
```

For more information about the network software for the IRIS workstation, see the manuals *IRIS Communications Guide*, *TCP/IP User's Guide, Version 2.0*, and the *IRIS XNS User's Guide*.

4.3.3 Using Multiple Kernels

The UNIX operating system is read into memory from a disk file at boot time. The file can have any name, but there are two names that are recognized in special ways: */defaultboot* and */vmunix*. The PROM monitor recognizes the name *defaultboot* in the root directory of the disk, and boots that file when no name is explicitly given. Certain UNIX commands recognize the name *vmunix* and expect it to contain a copy of the kernel currently running in memory.

New kernels are shipped in a directory called */kernels*, with names such as *3000.tcp* (the standard kernel), *3000.xns* (the XNS optional kernel), and *3000.nfs* (the NFS optional kernel). A copy of one of the kernels is installed as */defaultboot* and linked to */vmunix*. (A link is a method of giving the same file two different names — see *cp*(1)).

When a software option includes a new kernel (for example the NFS option), the kernel is automatically installed as */defaultboot* and */vmunix*. The PROM monitor will boot that kernel by default, and the various kernel-dependent UNIX commands find the image of the kernel in the */vmunix* file. However, some sites switch among more than one kernel, to have access to more than one communications option. In such an environment, it is necessary to copy the kernel being used to */vmunix*, and link it to */defaultboot* (This is done as the last step before taking the system down with the old kernel to reboot it with the new choice.) The *kernel*(1M) program takes care of these details. The argument is the short name of the kernel you want to install, e.g. *tcp*, *xns*, *nfs*, and so on.

Support for the IEEE 488 and IBM communications options are included in the TCP/IP and XNS kernels. Only the superuser can run this program.

4.4 Backups

When you install your IRIS workstation, make sure you have a *bootable backup tape* containing the stand-alone software, the root file system, and the user file system. You should receive a bootable backup tape with your workstation. Make a new bootable backup tape after you install new software or change the size of the swap partition on your disk (see Section 4.4.1).

After the workstation is in use, it is very important to make *periodic backup tapes* of the entire disk or the user data stored on the disk (see Section 4.4.2).

The IRIS workstation supports two tape drives: a quarter-inch cartridge tape drive and a half-inch tape drive. You can use the cartridge tape drive to make bootable backup tapes and periodic backup tapes, and for installing software updates. You can also boot the IRIS workstation from the cartridge tape drive in case the root file system on the disk is damaged (see Section 4.5.2).

You can use the half-inch tape drive to make periodic backups and to transfer files, but you cannot use it to boot the workstation.

Use the standard UNIX archiving tools, *tar*(1) or *cpio*(1), to read from and write to tapes. These two different tools perform essentially the same function; however, they are *not* interchangeable. If you create a backup tape using *tar*, you can only read that information back from the tape using *tar*.

4.4.1 Making a Bootable Backup Tape

You can use the bootable backup tape to boot the workstation and rebuild the disk in case the file system on the disk is damaged. Make a bootable backup tape as soon as you install a new workstation, and whenever you install new software or change the size of the swap partition on your disk.

You can make a bootable backup tape only if you have a quarter-inch cartridge tape drive.

NOTE: A bootable backup tape is dependent on the hardware and software configuration of the IRIS; you must make a bootable backup tape for each model you have. You cannot make a bootable backup tape on an IRIS 3030 and use it to recover an IRIS 3020.

To make a bootable backup tape, follow these steps:

1. Become the superuser. Make sure no other users are logged in, and reboot the system into single-user mode:

   ```
   su
   who
   sync
   reboot
   b
   fsck
   ```

   fsck will tell you if there are any problems with your file system. If there are problems, *fsck* will ask if you want to fix your system. Type *y* after all of its questions.

2. Change your working directory to the root file system:

   ```
   cd /
   ```

3. Mount the */usr* file system:

   IRIS 2500T:

   ```
   mount /dev/ip0c /usr
   ```

   IRIS 3020, 2400T:

   ```
   mount /dev/md0c /usr
   ```

IRIS 3030:

```
mount /dev/si0f /usr
```

4. Check the size of your root and user filesystems:

```
df
```

   Add up the numbers in the *use* column. If the total is more than 40,000, you must use the 60-megabyte tape to store both the root (/) and user (*/usr*) files on one tape. You can also store the root files on one tape using the `mkboot` command (see below) and store the user files on another tape using *tar* or *cpio* (see Section 4.4.2).

5. Put a new tape in the tape drive.

6. Rewind the tape:

```
mt rewind
```

7. Run the *mkboot* program. (If you have already backed up */usr* onto a separate tape, you don't need to use the `/usr` argument.) This will take approximately 30 minutes.

```
mkboot /usr
```

8. Make sure that your files were copied onto the tape:

```
tar  tv           # 1/4-inch cartridge tape device
tar  tv3          # half-inch tape device
cpio  -ihumd1     # 1/4-inch cartridge tape device
cpio  -iBtv3      # half-inch tape device
```

   If you get an error message instead of a list of the files you copied, rewind the tape with the command mt rewind, and try step 7 again. If you still get an error message, call the Geometry Hotline. You may have a bad tape drive.

See Section 4.5.2 for the procedures for rebuilding a damaged file system from the bootable backup tape.

4.4.2 Making Periodic Backups

This section describes the procedure for making routine backups of the IRIS workstation disk(s). You can back up the disk(s) to a tape on a local tape drive or across a network to a tape on another workstation.

Each file system is backed up onto a separate tape. Table 4-4 below gives the capacity of various tapes. The *df* command tells you how much of the file systems are used. The number in the use colume tell you how many kilobytes of information the tape must be able to store.

Tape Width	Tape Length	Approx. Capacity
1/4-inch cartridge	450 ft.	40 Mb
1/4-inch cartridge	600 ft.	60 Mb
Half-inch (PE format)	2400 ft.	44.6 Mb

Table 4-4: Approximate Tape Capacities for Backups

Cartridge Tape

Complete Backup
`tar c .`
`cpio -oha1 .`

Incremental Backup `find . -mtime -7 -print	tar c -` `find . -mtime -7 -print	cpio -oha1`

Read Tape
`tar x`
`cpio -ihumd1`

List Tape Contents
`tar xv`
`cpio -ihtv1`

Half-inch Tape

Complete Backup
`tar c3 .`
`cpio -oBa3 .`

Incremental Backup `find . -mtime -7 -print	cpio -oBa3`

Read Tape
`tar x3`
`cpio -iBumd3`

List Tape Contents
`tar tv3`
`cpio -iBtv3`

Table 4-5: Sample Tape Drive Commands

Table 4-5 lists samples of the UNIX commands for making periodic backup tapes. (See *find*(1) for more information on the arguments to *find*.) A *complete backup* backs up all files in a particular directory. An *incremental backup* backs up all files in a directory that have been modified in the last seven days.

CAUTION: If you have a quarter-inch cartridge tape drive, do not use multiple tapes to copy a file system that is too large to fit on one tape. Inserting a second tape, even when you are prompted to do so, could cause a system crash. If more than one tape is required, back up subdirectories of the file system onto separate tapes.

Issue these commands when you are in the directory that you want to back up. To back up a directory other than your own, you must be the superuser. For example, if you wanted to do a complete backup of the directory */usr* by copying it to a quarter-inch cartridge using *tar*, you would type:

```
su
cd /usr
tar c .
```

In general, you should periodically make backup tapes of everything on the disk.You may want to back up parts of the file system that change frequently, such as the user files. To simplify this process, you can store files that change most frequently in one directory, so they can be backed up independently. For example, you can store most user files in the directory */usr/people* and back it up daily.

Each *tar* or *cpio* command generates a single output file, which is written to tape. To put several of these output files on one tape, use tape devices that do not automatically rewind.

NOTE: If you interrupt the non-rewind cartridge tape device on an IRIS 3020 or 2400T, you must reboot the system to use the tape drive again. To avoid this problem, remove the cartridge from the drive to stop the device.

To use the no-rewind tape devices, replace the *tar* and *cpio* commands in steps 4 and 8 below with these lines:

```
tar c2 .        # non-rewind cartridge tape device
tar c4 .        # non-rewind half-inch tape device
cpio -oh2       # non-rewind cartridge tape device
cpio -oh4       # non-rewind half-inch tape device
```

Backing Up a System with a Local Tape Drive

To back up the disk on a workstation with a local tape drive, follow these steps:

1. Become the super-user:

   ```
   su
   ```

   If you are only backing up the */usr* file system, go to step 6.

2. Reboot the system into single-user mode, and check the file system:

   ```
   who
   sync
   reboot
   b
   fsck
   ```

   Correct any errors reported by *fsck*.

3. Put a tape into the tape drive.

4. Back up the root file system using one of these commands:

   Using tar:

   ```
   tar c .          # rewind cartridge tape device
   tar c3 .         # rewind half-inch tape device
   ```

Using cpio:

```
cpio -oh1 .            # rewind cartridge tape device
cpio -oh3 .            # rewind half-inch tape device
```

5. Verify that the files you copied are indeed on the tape:

```
tar tv        or       cpio -ihtv1
tar tv3                cpio -iBtv3
```

   If they were copied successfully, you will see a list of files scroll by. If you get an error message instead of a list of the files you copied, issue the command `mt rewind` and try to back up the files again. If this still doesn't work, call the Geometry Hotline. You may have a bad tape drive.

6. Mount the */usr* file system:

   IRIS 2500T:

```
mount /dev/ip0c /usr
```

   IRIS 3020, 2400T:

```
mount /dev/md0c /usr
```

   IRIS 3030:

```
mount /dev/si0c /usr
```

7. Change directories to */usr*:

```
cd /usr
```

8. Put another tape into the tape drive.

9. Back up the */usr* file system using one of these commands:

   Using tar:

```
tar c .             # rewind cartridge tape device
tar c3 .            # rewind half-inch tape device
```

   Using cpio:

```
cpio -oh1 .         # rewind cartridge tape device
cpio -oh3 .         # rewind half-inch tape device
```

10. Verify that the files you copied are on the tape.

```
tar tv      or      cpio -ihtv1
tar tv3             cpio -iBtv3
```

    If they were copied successfully, you will see a list of files scroll by. If you get an error message instead of a list of the files you copied, issue the command `mt rewind` and try to back up the files again. If this still doesn't work, call the Geometry Hotline. You may have a bad tape drive.

11. Mount any additional file systems and back them up following the same procedure.

Backing Up a System without a Local Tape Drive

You can back up an IRIS workstation with no tape drive across a TCP/IP or XNS Ethernet onto a tape on another workstation. Both systems must have network communication capabilities. The system with a tape drive must have a host name, which is referred to in this description as *hostA*, and must be in multi-user mode.

To back up a disk onto a tape installed in a remote workstation, follow the procedure for backing up a system with a local tape drive in the previous section, but use the command lines below for steps 4 and 9. Replace *rmt1* with *rmt2*, *rmt3*, or *rmt4* depending on your backup device (see Table 4-1).

4. Back up the root file system:

   Using tar and TCP/IP:

   ```
   tar caBf - . | rsh hostA dd ibs=10k obs=200k of=/dev/rmt1
   ```

   Using tar and XNS:

   ```
   tar caBf - . | xx hostA dd ibs=10k obs=200k of=/dev/rmt1
   ```

   Using cpio and TCP/IP:

   ```
   cpio -oha . | rsh hostA dd ibs=10k obs=250k of=/dev/rmt1
   ```

   Using cpio and XNS:

   ```
   cpio -oha . | xx hostA dd ibs=10k obs=250k of=/dev/rmt1
   ```

9. Back up the *usr* file system:

   Using tar and TCP/IP:

   ```
   tar caBf - . | rsh hostA dd ibs=10k obs=200k of=/dev/rmt1
   ```

   Using tar and XNS:

   ```
   tar caBf - . | xx hostA dd ibs=10k obs=200k of=/dev/rmt1
   ```

   Using cpio and TCP/IP:

   ```
   cpio -oha . | rsh hostA dd ibs=10k obs=250k of=/dev/rmt1
   ```

   Using cpio and XNS:

   ```
   cpio -oha . | xx hostA dd ibs=10k obs=250k of=/dev/rmt1
   ```

4.5 Crash Recovery

Depending on the cause, a system crash can require some software repair. This section describes the steps you should take immediately after a workstation crash and the procedures for repairing the disk if the software is damaged.

See Appendix B for a list of system error messages and their probable causes. Also, see the *crash*(8) manual entry.

If the IRIS workstation stops running, follow this recovery sequence:

1. Try to reboot the workstation:

   ```
   su
   who
   sync
   reboot
   b
   ```

2. If the system does not respond, wait one minute, and press the `Reset` button located on the back panel of the cabinet.

   The system should attempt to follow the normal boot sequence.

3. Boot the system in single-user mode and check the file system.

   ```
   b
   fsck
   ```

4. *fsck* may identify problems with the file system (see Section 3.3 and *fsck* in the *UNIX Programmer's Manual, Volume II*).

5. If *fsck* runs without any problems, enter multi user mode:

   ```
   multi
   ```

If rebooting and running *fsck* does not solve the problem, see the procedures below for identifying the problem and recovering the system.

4.5.1 Critical UNIX Files

Accidental changes to the files critical to UNIX operation can prevent the UNIX operating system from booting. Table 4-6 lists some of the files that are most likely to cause serious problems.

Critical Files	
File	*Purpose*
/	Root directory
/vmunix	Kernel
/defaultboot	Kernel used by default in booting
/dev	Device directory
/dev/console	Console tty device
/dev/syscon	System console device
/dev/systty	System tty device
/dev/md0a	Root file system device
/dev/rmd0a	Raw root file system device
/dev/drum	Paging device for virtual memory
/etc	Miscellaneous file directory
/etc/init	Program that starts other user processes
/etc/inittab	Table for *init*
/etc/passwd	Table of user login names, passwords, and shells
/bin	Commands directory
/bin/su	Used to set environment before *exec*ing shell
/bin/sh	Bourne shell
/bin/csh	C shell

Table 4-6: Files Critical to UNIX Operation

The following subsection describes how to recover a system that does not boot. You use a bootable backup tape to load a new root file system onto the disk. This procedure deletes any data that was on the root file system of the disk.

4.5.2 Recovery by Booting from Tape

If the root (/) or user (*/usr*) file system on the workstation's disk is damaged, you can rebuild the system from a bootable backup tape made for this purpose (see Section 4.4.1). You can rebuild the system from tape only if you have a quarter-inch cartridge tape drive. This procedure writes over the contents of the disk.

You must use a tape made on the same IRIS model that you are trying to recover, i.e., you cannot use a tape made on an IRIS 3020 to recover an IRIS 3030.

If the bootable backup tape does not contain the most recent user file system, you should update the user file system from the periodic backup tapes after rebuilding the disk.

To restore a disk from a backup tape, follow these steps:

CAUTION: You must follow all instructions exactly when you use the *fex* programs. Failure to follow instructions could result in serious damage to your file system.

1. Set configuration switches 1 through 5 for booting from the PROM monitor (see Table 3-1).

2. Reset the workstation by pressing the `Reset` button. The IRIS displays the PROM monitor prompt:

```
iris>
```

3. Put the bootable backup tape in the tape drive.

4. Boot the *mdfex*, *ipfex* or *sifex* program.

   IRIS 2500T:

   ```
   b ct0: mdfex
   ```

   IRIS 3020, 2400T:

   ```
   b ct0: mdfex
   ```

   IRIS 3030:

   ```
   b ct0:sifex
   ```

   The *mdfex*, *ipfex* or *sifex* initial screen and prompt appear.

   ```
   SGI Formatter/Exerciser Version :x Date
   Drive: 0, Name: name, Serial: serial number
   ```

   IRIS 3020:

   ```
   mdfex x>
   ```

   IRIS 3030:

   ```
   sifex x>
   ```

   NOTE: If an error message appears instead, call your local System Engineer or the Geometry Hotline (see Chapter 1).

5. Instruct *mdfex*, *ipfex* or *sifex* to create the root file system by entering the *t* command:

```
t
```

6. Accept the *mdfex*, *ipfex* or *sifex* defaults for tape file and unit numbers by pressing RETURN after each of these prompts:

```
Tape file (2)?
Unit (0)=
```

7. Confirm file system *a* by pressing RETURN in response to the file system prompt:

```
File System (a)?
```

   mdfex/*sifiex* displays a message about the block sizes it will use for the copy. When it is ready it displays the message:

```
Type 'go <return>' to begin ...
```

8. Type:

```
go
```

   then press RETURN.

The system displays these messages:

```
Started
rewinding...
*** Copy started...
1 2 3 4....
Tape to Disk Copy complete
```

Thn you see the system prompt.

IRIS 3020, 2400T:

```
mdfex x>
```

IRIS 3030:

```
sifex x>
```

9. Return your configuration switches on the cabinet back to their previous positions.

10. Re-enter the PROM monitor by entering the quit command:

    ```
    q
    ```

    Confirm your intention to quit by entering `y`.

11. Boot the disk:

    ```
    b
    ```

    UNIX should come up in single-user mode. Run fsck to check your files. If the user file system was not damaged, go to step 15.

    If the */usr* file system suffered major damage, perform the remaining steps in this section to restore it.

    If you made a separate backup tape for */usr*, remove the bootable backup tape and put the */usr* tape in the tape drive.

12. Make a clean user file system:

    IRIS 2500T:

    ```
    mkfs /dev/rip0c
    labelit /dev/rip0c usr sgi
    ```

    IRIS 3020, 2400T:

    ```
    mkfs /dev/rmd0c
    labelit /dev/rmd0c usr sgi
    ```

    IRIS 3030:

    ```
    mkfs /dev/rsi0f
    labelit /dev/rsi0f usr sgi
    ```

13. Mount the new */usr* file system and change to that directory:

    IRIS 2500T:

    ```
    mount /dev/ip0c /usr
    cd /usr
    ```

    IRIS 3020, 2400T:

    ```
    mount /dev/md0c /usr
    cd /usr
    ```

    IRIS 3030:

    ```
    mount /dev/si0f /usr
    cd /usr
    ```

14. Rewind the tape and copy */usr* from the tape to the disk.

    If you have /usr on its own tape, do not issue the command `mt fsf 2` that is shown below. This command positions the tape to the third file on it, which is where */usr* is located when you put both the root (/) and user (*/usr*) file systems on the same tape. Use this command only if you made your bootable backup tape with the command *mkboot /usr*.

    ```
    mt rewind
    mt fsf 2
    cpio -ivhmud2
    ```

15. To recover the root and user file system, use any backup tapes that are more current than the bootable backup tape.

5. IRIS 3010 System Administration

This chapter contains operation and configuration procedures for the IRIS 3010. Topics covered are shown in the table below. At this point, it is assumed that you have installed and booted the IRIS (see Chapters 2 and 3).

- Installing the host software
- Connecting to the host
- Issuing commands to the terminal emulator
- Running a demonstration program
- Shutting down the IRIS terminal
- Recovering from a boot failure
- Configuring the startup environment
- Configuring XNS
- Configuring TCP/IP
- Configuring miscellaneous UNIX files

5.1 Installing the Host Software

This section contains instructions for installing the host-resident IRIS 3010 support software in a VAX UNIX 4.2BSD environment. The entry name for the workstation host-resident software is *host*. Instructions for installing host software in other environments are shipped with the software.

The software is delivered on a 1600 bpi half-inch magnetic tape in *tar*(1) format.

The software can be stored anywhere on the host file system. The procedure below assumes that the distribution software is installed in a directory called

/usr/iris. (Before doing this procedure, install and boot your IRIS workstation.)

To install the host software, follow these steps:

1. Log in to UNIX from a terminal that is already attached to the host.

2. Create a directory to hold the distribution software:

```
mkdir /usr/iris
```

3. Change to the directory you just created:

```
cd /usr/iris
```

4. Load the distribution tape onto a tape drive and read the distribution software into the new directory, using the *tar* command:

```
tar xv
```

5. Add the *TERMCAP* entry for the IRIS Terminal in */usr/iris/c/iris.termcap* to the terminal description data base in */etc/termcap*. (See the *termcap*(4) manual page in the *UNIX Programmer's Manual Volume IB* for more information on *TERMCAP*)

6. Change directories to */usr/iris/c*:

```
cd /usr/iris/c
```

7. Compile the C language remote host Graphics Library (*librgl2.a*), the IRIS 3010 configuration tools, and the demonstration programs, using the */usr/iris/c* makefile:

```
make all
```

8. To install the library in */usr/lib* execute:

```
make install
```

9. If a FORTRAN language remote host Graphics Library (*libfrgl2.a*) is desired, change directories to */usr/iris/f77* and compile it:

```
cd /usr/iris/f77
make all
```

   The FORTRAN demonstration programs are also compiled.

   To install the library in */usr/lib*, execute:

```
make install
```

   To install the library in */usr/lib* on an IRIS, execute:

```
make demos
```

For a terminal with network communication capabilities, the host must also be equipped with the network communication support software and hardware. See *VAX 4.2BSD UNIX XNS Release N1.0 Installation Guide* for the hardware and software installation procedures for a host running 4.2BSD UNIX.

5.2 Operation

The IRIS 3010 is configured to run automatically a terminal emulation program, *wsiris*(1), after booting. It causes the IRIS to behave like a regular character terminal, with two additional capabilities:

- Setting up communication with the host.
- Interpreting graphics command sequences from the host.

The terminal emulator program can execute an interactive subshell from which UNIX commands can be issued. A complete UNIX operating system is not provided on the IRIS 3010.

It includes only those commands necessary to boot and perform basic system administration.

With the IBM option installed, the IRIS 3010 runs a different terminal emulator, *t3279*(1). Its user interface is completely different from the one described below.

5.2.1 Connecting to the Host

This section describes the terminal emulator interface used to connect with the host. (See Section 3.2 for the procedures for booting the terminal emulator program. See Section 5.3.2 for the procedure for configuring the terminal emulator program for the needs of a specific host. See Section 5.2.2 for the commands recognized by *wsiris*.

When *wsiris* first comes up on the IRIS, it displays the startup message:

```
IRIS GL2 Terminal Emulator
```

The terminal then prompts for instructions regarding the connection to the host. The prompt varies, depending on the configuration of the system.

IRIS 3010 with XNS Ethernet

An IRIS 3010 with no Ethernet controller, with only XNS Ethernet communication software, or with previously configured TCP/IP communication software prompts for the name of the host computer:

```
Connect to what host?
```

To establish communication over the serial line, enter `serial`.

To establish communication over an Ethernet, enter the name of the host.

For Ethernet communication, the host must have network support software installed. For XNS communication, the host must be running the *xnsd*(1M) daemon. For TCP/IP communication, the host must be running the *telnetd*(1C) daemon. If you do not know the name of a UNIX host, enter the *hostname*(1) command on a terminal that is already logged onto such a host.

The IRIS establishes a connection with the host, which should display its usual login prompt.

IRIS 3010 with TCP/IP

An unconfigured IRIS 3010 with TCP/IP communication capabilities prompts for the Internet addresses of the IRIS and the host:

```
Enter IRIS IP address:

Enter host IP address:
```

It accepts only Internet addresses consisting of four numbers separated by periods (*n.n.n.n*).

To convert a two-field Internet address to the correct format, insert two fields of zeros in the middle of the address. For example, if the representation of a host name in the */etc/hosts* table on a 4.2BSD UNIX system is 42.2, enter the address as 42.0.0.2.

The IRIS establishes a connection with the host, which should display its usual login prompt.

Once communication with the TCP/IP host has been established, you can copy an Internet host table onto the terminal's hard disk (see Section 5.3.4). You can then connect to an TCP/IP host by simply entering the name of the host when prompted. See the documentation that accompanies the communications software.

5.2.2 Issuing Commands to the Terminal Emulator

After a connection with the host has been established, the terminal emulator program recognizes some local commands, which are stripped from the data stream and processed locally instead of being transmitted to the host. The commands begin with ~.

To issue a command to the terminal emulator, enter the escape character at the beginning of a line, followed by the rest of the command.

The character prior to ~ must have been a RETURN or a LINEFEED. *wsiris* examines the next few characters to see if they are any of the following commands:

~~ Send a single '~'.

~. Force *wsiris* to exit. Ethernet connections are closed, but serial connections may remain open.

~! Escape to a C shell on the IRIS 3010. The shell allows the user to run programs from the local disk, such as the flight simulator:

```
~demos/flight
```

If you are familiar with the UNIX editor *vi*, you can edit the configuration files from the local C shell.

To return to communication with the remote host, exit the C shell by entering the `exit` command.

~! can also be entered in response to the hostname prompts before a connection is established.

~DEL Reboot the IRIS with */etc/reboot*. You are asked for confirmation. This command succeeds only for the super-user.

~BREAK If using serial communications, send a break character.

~%D*n* Toggle the value of the **–d** *n* debugging option (see Section 5.3.2). **~%D1** interactively turns the writing of the log file on and off. When you turn logging on, log output is appended to the current log file. If *n* is **3**, the new logfile name consists of all characters following the **3** until terminated by a RETURN. If logging is on, the current log file is closed and the newly named one is opened. Subsequent log output is appended to it. If you specify a null file name, the name of the current log file is displayed. **~%D2** is ignored.

~%M*n* Set the monitor type to *n*. *n* has one of the following values:

0	30Hz
1	60Hz
2	NTSC
3	50Hz
9	PAL

~%P Toggle the value of the **–p** option (see Section 5.3.2). The **–p** option controls whether textport output is printed when the textport is off.

~%R Reset the display by forcing a `ginit` and a `tpon` to be executed. This will load the default color map and turn on the textport. If the window manager is running, the terminal also executes any `mapcolor` commands stored in the window manager configuration file (see Section 5.3.2).

~%S*speed* Change the serial communications baud rate to *speed*. The parameter *speed* can have these values:

75 110 134 150 300 600 1200 2400 4800 9600 19200

The default is 9600 baud.

~%T Toggle the textport. That is, turn the textport on if it is off, and off if it is on.

The textport does not automatically disappear when it is turned off. If the remote system isn't echoing input, especially after a graphics program has been interrupted, the textport has been turned off. Issue this command to turn it back on.

~%U Unlock the keyboard. Whenever the IRIS is accepting graphics commands, the keyboard is locked and any input from it is thrown away. Use this command to force characters through to the remote system, such as the interrupt character to kill the remote program. The keyboard will be locked again at the next switch back into graphics mode.

~%X Toggle the value of the **–x** option (see Section 5.3.2). This option determines whether the handling of XON (CTRL-q) and XOFF (CTRL-s) is done locally.

~%Z*n* Toggle the value of the **–z** *n* option (see Section 5.3.2). Currently, the only valid value for *n* is **1**.

The escape character can be changed from ~ with the **–e** option specified in the *wsiris* configuration file (see Section 5.3.2). If you specify **–e none**, there is no escape character and none of the above interpretations occur. In this case, if the window manager is not running, there is no way to force *wsiris* to exit.

If the remote program queues the keyboard, all keystrokes will normally accumulate in the event queue. To divert them to *wsiris*—for example, to issue one of the ~ commands described above—hold down the SETUP key while typing. The events from individually queued keyboard keys are also not placed in the queue while SETUP is down. See the *IRIS User's Guide* for further information on queuing the keyboard and the event queue.

5.2.3 Running a Demonstration Program

After booting a new IRIS 3010 and connecting to a host, you should test the system by running a demonstration program. The hard disk that comes with the IRIS 3010 contains a non-bootable copy of the *flight* demonstration program. To run *flight* from the hard disk, you must first boot UNIX on the terminal, then escape into the C shell and run the program from the shell.

1. Boot the terminal from the hard disk, as described in Section 3.2.

2. When the *wsiris* terminal emulator is running, enter the sequence ~! to enter the C shell:

   ```
   Connect to what host? ~!
   ```

   The terminal displays the C shell prompt:

   ```
   %
   ```

3. To run the demonstration program, enter the pathname:

```
~demos/flight
```

The first screen of the *flight* program appears.

4. To leave the *flight* program, press the `ESC` key.

The IRIS again displays the C shell prompt.

5. To return to the terminal emulator program, enter the *exit* command:

```
exit
```

5.2.4 Shutting Down the IRIS Terminal

To shut down an IRIS 3010 running *wsiris*:

1. Issue the ~`DEL` command.
2. Set the `Power` switch on the front of the cabinet to the `Off` position.

5.2.5 Recovering from a Boot Failure

If a critical file on the IRIS 3010 disk becomes corrupted, the boot procedure fails and the terminal emulator does not run. If this happens, use the procedure in this section to restore the file system from the copy stored on the backup partition of the disk.

1. Turn control of the IRIS over to the PROM monitor by setting configuration switches 1 through 5 to OCOCC and resetting the terminal (see Section 3.4).

2. Boot the recovery system:

```
iris> b md0c:recover
```

This will boot a copy of the UNIX kernel on the backup partition, and run a program to copy the backup partition onto the normal operating partition. After this program is finished, it returns control of the IRIS to the PROM monitor, which displays its prompt:

```
iris>
```

3. Boot the terminal normally from the local disk (see Section 3.2).

4. Repeat the configuration sequence described in Section 5.3 .

If this procedure fails to return the IRIS to its former working state, contact Silicon Graphics field service personnel through the Geometry Hotline number listed in Chapter 1.

If you are familiar with UNIX, you can use a 9600 baud ASCII terminal connected to `Port 2` of the IRIS 3010 to log in as *root*. This may be useful for diagnostic purposes in case the standard console textport is not functioning properly. To connect an ASCII terminal to the IRIS, see Section 7.1.

5.3 Configuring the Software

The IRIS 3010 software is based on the same System V UNIX kernel used on the IRIS 3020 and 3030. IRIS 3010 users do not necessarily interact with UNIX because the terminal emulator program running on top of UNIX provides an interface almost identical to the one on IRIS terminals. The terminal emulator interface is described in Section 5.2.

Because the IRIS 3010 is UNIX-based, the IRIS window manager runs on it. The window manager is described in the *IRIS User's Guide, Versions 2.1* or later.

An IRIS 3010 can have both XNS and TCP/IP Ethernet communication capabilities. Configuration of the IRIS 3010 for network communication is not necessary for normal operation, but some configuration can simplify the process of communicating over the network and can increase the terminal's communication capabilities.

5.3.1 Configuration Tools

A number of files stored on the IRIS 3010's hard disk allow for three kinds of configuration:

- Startup environment configuration (see Section 5.3.2).
- Network communication configuration (see Sections 5.3.3 and 5.3.4).
- Miscellaneous UNIX configuration (see Section 5.3.5).

The following subsections describe these files in detail. Samples of the configuration files are duplicated in the host software distribution.

If you are familiar with UNIX and its utilities, you can edit these files locally on the IRIS 3010, using the *wsiris* shell escape command (~!) and the *vi* editor. If you prefer, you can edit the files on the host, using whatever text editor is most familiar to you, then transfer the files to the disk on the IRIS 3010. This strategy has the advantage of leaving backup copies of the customized configuration files on the host.

To support configuration of the IRIS 3010 from a host, the host software distribution includes two utility programs: *iftp* and *irsh*. Both programs are available when you are logged on to the remote host from the IRIS 3010. These commands are run from the host while the IRIS 3010 is running wsiris.

File Transfer Program

The *iftp* program transfers files between the host and the IRIS 3010. The command takes three arguments:

`iftp` *mode hostfile 3010file*

Mode has the following values:

w Write *hostfile* to *3010file*. A temporary file on the IRIS 3010 is written to and then renamed.

o Overwrite. It is the same as **w**, but the temporary file is not made.

r Read from *3010file*, write to *hostfile*.

3010file must be an absolute pathname (i.e., it must start with /).

Local Execution Program

The *irsh* program issues commands to the UNIX operating system running on the IRIS 3010 from the host. The program takes the IRIS 3010 UNIX command and arguments as its arguments:

`irsh` *3 010cmd arg ...*

The main uses of *irsh* are renaming and removing files on the IRIS 3010 disk.

To rename the file *oldname* to *newname*, enter this sequence:

`irsh mv` *oldname newname*

To remove *file*, enter:

`irsh rm` *file*

Configuration from a Unix Host

After the distribution software has been installed on the UNIX host, you can use this procedure to configure the IRIS 3010 from the host:

1. Boot the IRIS 3010 from its local disk and log in to the host (see Section 3.2).

2. Change directories to */usr/iris/c*.

```
$ cd /usr/iris/c
```

3. Edit the configuration files as described in the remainder of this chapter.

4. Transfer the files to the IRIS 3010 by invoking the *iftp* program for each file. To transfer all files in a single step, use the makefile argument *install-config-files*.

```
$ make install-config-files
```

5.3.2 Configuring the Startup Environment

You can configure the startup environment on the IRIS 3010 in the following ways:

- Custom-configuring the terminal emulator program, *wsiris*.
- Causing the window manager to start automatically during bootstrap.
- Installing your own startup program.

The next three sections describe these procedures.

Terminal Emulator Configuration

The terminal emulator program, *wsiris*, is customized through the file *.wsirisrc*. The working copy of *.wsirisrc* is stored on the IRIS 3010 hard disk. The file contains any number of the option flags listed below, separated by whitespace (spaces, tabs, and newlines), and possibly followed by a hostname. Arguments to individual option nags are separated from their options by whitespace.

The file may contain any of these options:

–d *n* Debug. You may use multiple **–d** options. The legal values for *n* are:

1 Generate a logfile, named */LOGFILE.0*, of the communication between the IRIS 3010 and the host. Graphics commands and data are interpreted and written to the logfile as a command name and a series of arguments. The arguments are shown as *type=value*, where *type* is a single character representing the argument type:

a, A	array
b, B	byte (8 bits)
c, C	character string
f, F	float (32 bits)
l, L	long (32 bits)
s, S	short (16 bits)
o	logical (8 bits)
O	logical (32 bits)

If the character is capitalized, then the the argument was sent by the IRIS, otherwise it was sent by the remote system. The *value* of array types is always *array*.

3 Use the next argument for the name of the logfile created by the **–d 1** option in place of */LOGFILE.0*.

–e *c* Set the escape character from the default ~ to *c*. Lines beginning with the escape character can be commands to *wsiris*. If *c* is **none**, there will be no escape character.

–h Use half-duplex serial communications. The default is full-duplex.

–i Try the TCP/IP communication protocol first. When using Ethernet communications in the absence of the **–i** option, the system first attempts a connection to *hostname* using XNS protocols. If that fails, it attempts a connection using TCP/IP.

–l *line* Use device *line* for the serial connection. The device names of the three available serial ports of the IRIS 3010 are:

Port 2	ttyd1
Port 3	ttyd2
Port 4	ttyd3

The default *line* is ttyd2.

–p Print textport output even when the textport has been turned off. By default, after the receipt of a `tpoff` command from the remote system, all textport output is thrown away until a `tpon` is received. (See the *IRIS User's Guide* for information on `tpoff` and `tpon`.) The **–p** option inhibits this, and characters written to the textport will appear when it is turned on again.

–s *speed* Use *speed* baud for serial communications. The legal values for *speed* are:

75 110 134 150 300 600 1200 2400 4800 9600 19200

The default is 9600.

–x Enable local XON/XOFF flow control. Normally, when read from the standard input, the characters XON (`CTRL-q`) and XOFF (`CTRL-s`) are sent to the remote system. Specifying this option causes them to be used locally to stop and restart the display of text. When using serial communications, this option also enables the use of these characters by the serial driver of the IRIS for automatic flow control of data being received from the host. This automatic flow control is normally disabled as it will not work correctly if there are two independent sources of these characters.

–y If using serial communications, interpret XON and XOFF characters from the remote host as output flow control commands. By default, they are treated as ordinary data. You must not give this flag if you are using eight-bit (fast mode) communications.

–z *n* Special instructions. Multiple **–z** options can be given. The legal values for *n* are:

1 Interpret `ginit` commands as `gbegin` commands. (See the *IRIS User's Guide* for information on `ginit` and `gbegin`.)

2 Execute a `ginit` immediately upon startup. This is useful when *wsiris* has had to be restarted and the remote program is still sending graphics commands.

3 When attempting a connection using Ethernet communications, try only one network protocol instead of both. TCP/IP is tried if the **–i** flag is given, and XNS if it isn't.

4 Emulate a Tektronics 4010.

You can specify a hostname following the option flags. This causes *wsiris* to connect to this host and not give a prompt:

```
Connect to what host?
```

The IRIS 3010 is shipped with a default */.wsirisrc* file that contains these options:

```
-z 3 -x
```

The host distribution contains a copy of this file named *wsirisrc*. To configure the IRIS 3010 from the host, edit the host copy of the file, then transfer it to the IRIS with this command:

```
iftp w wsirisrc /.wsirisrc
```

The default configuration file may not work for some serial communications environments, e.g., if half-duplex is required. In this case, */.wsirisrc* must be edited locally on the IRIS 3010.

If an invalid */.wsirisrc* is put on the IRIS 3010 disk, it will not boot correctly. If this happens, use the boot failure recovery procedure described in Section 5.2.5.

Window Manager Configuration

The IRIS 3010 can be configured to start the window manager, *mex*, automatically. The window manager options are controlled through the file */.mexrc*. The *IRIS User's Guide* contains a description of what *mex* expects to find in this file. The default configuration file on the IRIS 3010 is different from the one described in the *IRIS User's Guide*.

The boot sequence checks for the existence of */.mexrc*. Only if */.mexrc* is found will *mex* be started. The IRIS 3010 is shipped with the default window manager configuration file with the name */mexrc*, which means that *mex* is not started up during bootstrap. To have *mex* start automatically, change the filename to */.mexrc* by entering this command on the host:

```
irsh mv /mexrc /.mexrc
```

To disable *mex* once it's been enabled, enter:

```
irsh mv /.mexrc /mexrc
```

The host distribution contains a copy of *mexrc*. After editing this file, you can copy it to the IRIS with this command:

```
iftp mexrc /.mexrc
```

Installing Your Own Startup Program

The program that the IRIS 3010 runs after booting can be changed from *wsiris* to a custom startup program. To do this, edit the file */bin/tesh* and find the line:

```
set termemul = /bin/wsiris
```

Replace `/bin/wsiris` with the pathname of the program you want to have run.

If the IBM option is installed, *t3279* is run instead of *wsiris*, and this line in */bin/tesh* must be changed:

```
set termemul = /bin/t3279
```

Replace `/bin/t3279` with the pathname of the program you want to have run.

If you add custom files to the root partition, you should duplicate them on the backup partition so that they can be restored in case of boot failure. Before copying these files, make sure that the IRIS boots and runs your startup program correctly. The disk contains a script, */etc/rootcopy*, which copies the entire root partition to the backup partition. Run *rootcopy*, specifying the argument *ok*:

```
rootcopy ok
```

5.3.3 XNS Configuration

Operation of the IRIS 3010 using XNS requires no configuration. The IRIS does not need a hostname, since it initiates all connections, and the XNS protocol does not require that both ends of a circuit have a name.

The terminal emulation program can be configured to initiate a connection to a single host automatically every time it is started up. This procedure is documented in Section 5.2.1.

5.3.4 TCP/IP Configuration

The TCP/IP option to the IRIS 3010 terminal is configured with the hostname file */etc/sys_id*, and the host tables */etc/hosts* and */etc/hosts.equiv*. The hostname file contains the hostname of the IRIS 3010. The information that goes into the host tables is described in the documentation that accompanies the communications software.

A sample hostname file, *sys_id*, is included in the host distribution. To give the IRIS 3010 a hostname, edit this file to contain the desired name, and then enter the command:

```
iftp w sys_id /etc/sys_id
```

Hostnames must not be more than eight characters.

Next, create the two host table files on the host. Then transfer them to the IRIS 3010 terminal disk. If, for example, the two files on the host are named *hosts* and *hosts.equiv*, enter these commands:

```
iftp w hosts /etc/hosts
iftp w hosts.equiv /etc/hosts.equiv
```

Until these files exist on the IRIS 3010, the terminal emulator always prompts for the Internet addresses of the IRIS and the host. Once the terminal emulator knows the local address, it prompts only for the host address.

The terminal emulator can be configured to initiate a connection with a single host automatically each time it is started up. This procedure is documented in Section 5.2.1.

If the IRIS 3010 disk contains an invalid */etc/hosts* file, it will probably not be able to communicate with the host.

To recover from this situation, use one of these strategies:

- Edit */etc/hosts* locally on the IRIS 3010.
- Transfer a valid copy from the host over the serial communications line.
- Use the boot failure recovery procedure described in Section 5.2.5 to start over.

5.3.5 Miscellaneous Unix Configuration

Time Zone

Several UNIX utilities need to know the time zone in order to display the time correctly. The time zone entry is contained in the file */etc/TZ*. The file has three fields:

1. A three-character abbreviation for the time zone.
2. The offset in hours of the time zone from Greenwich Mean Time.
3. A three-character abbreviation for the time zone when in daylight savings time. If daylight savings time is not used, this field should be omitted.

The IRIS 3010 is shipped with the time zone set for Pacific Standard Time:

```
PST8PDT
```

The file *TZ* in the host distribution contains a sample time zone file. After you edit it, to transfer it to the IRIS 3010 disk, execute:

```
iftp w TZ /etc/TZ
```

Automatic Date Setting

By default, the IRIS 3010 prompts you to check the date each time it is booted.

```
Enter the correct date (mmddhhmm[yy][.ss]):
```

If the system finds an executable file */etc/rc.getdate*, you are not prompted, and the result of its execution is used.

A sample version of this file named *rc.getdate* is included in the host distribution. It is a shell script that gets the date from an IRIS workstation designated to be the network date server. To use it, edit it and change "host" to be the name of the date server machine. Then execute:

```
iftp w rc.getdate /etc/rc.getdate
irsh chmod a+w /etc/rc.getdate
```

To disable the automatic date setting, remove */etc/rc.getdate*:

```
irsh rm /etc/rc.getdate
```

6. Disk Configuration

This chapter describes how to configure a second disk, how to change the swap partition on a disk, and how to make a set of floppy disk for backing up or rebuilding the disk.

This chapter covers these topics:

- Disk specifications
- IRIS workstation disk configuration
- IRIS terminal disk configuration

The disks on the IRIS 3010 and 3020 are called `md`; the disks on the IRIS 3030 are called `si`. To find out what brand of disk you have on your workstation, issue the *sgilabel* command:

IRIS 3010/3020:

```
sgilabel md0
```

IRIS 3030:

```
sgilabel si0
```

A first disk is disk zero; a second disk is disk 1. Disks are divided into a number of partitions depending on the brand and size; the partitions are labeled with lower-case letters **a** through **g**. The two formatter/exerciser programs, *mdfex* and *sifex*, correspond to the two types of disk, `md` and `si`. These programs enable you to change the size of the swap partition on your disk.

The commands *mkfs*(1M) and *labelit*(1M) enable you to configure a second disk.

In this procedure you create and label file systems, then modify configuration files so that the file systems are mounted and checked by *fsck*(1).

6.1 Disk Specifications

Table 6-1 shows the disks available on IRIS series 3000 models, the number of megabytes per cylinder on each disk, and the size of the partitions.

IRIS Model/ Disk Drive	Mb/Cyl.	Partitions	Cylinders	Mb
IRIS 3010		md0a	123	4.28
Tandon	.0348	md0b	360	12.53
		md0c	123	4.28
		md0g	606	21.1
IRIS 3020		md0a	150	9.14
Vertex	.0609	md0b	149	9.08
(Priam)		md0c	670	40.82
		md0g	969	59.04
Maxtor	.0696	md0a	150	10.4
		md0b	149	10.38
		md0c	586	40.8
		md0g	885	61.58
IRIS 3030		si0a	59	9.44
Fujitsu		si0b	100	16.00
or	.164	si0c	200	32.00
Hitachi		si0d	446	71.40
		si0f	646	103.40
		si0g	805	128.80

Table 6-1: Disk Partition Sizes

NOTE: The disk on your IRIS 3020 may appear to have more space than is shown below, but this space is not supported by Silicon Graphics' software.

On all disks, partition **g** represents the entire disk and can be used in place of, not in addition to, the other partitions. For the IRIS 3030 only, the **f** partition is a combination of the **c** and **d** partitions.

Table 6-2 contains estimates of the disk storage requirements for UNIX software, the IRIS Graphics Library, and options that are available for the IRIS. These estimates fluctuate slightly from one release to the next as new utilities and other software are added to the system.

Software Category	Space Required (Megabytes)
Total, minimum system:	16.61
On-line manual pages (*/usr/man/**)	2.90
Demonstration programs (*/usr/people/demos/**)	4.37
Games (*/usr/games/**)	1.29
Gifts (*/usr/people/gifts/**)	1.78
Graphics Library tutorial (*/usr/people/tutorial/**)	0.63
Mail	1.14
Compiler Options:	
FORTRAN compiler option	1.46
Pascal compiler option	0.52
Other Options:	
NFS communications option	2.19
XNS communications option	0.74
Troff laser printer option	2.42
Emacs text editor	3.10
Total with all options installed:	39.15

Table 6-2: Disk Storage Requirements

6.2 IRIS Workstation Disk Configuration

The three subsections below describe these procedures:

- Changing the swap partition size on an IRIS 3020
- Changing the swap partition size on an IRIS 3030
- Configuring a second disk on an IRIS 3020/3030

If you have an IRIS 3010, see Section 6.3.

On an IRIS 3020 or 3030, disk zero is divided into three partitions: the root partition, which contains the root file system; the swap partition, which is used by UNIX for paging; and the user partition, which contains the */usr* file system. On an IRIS 3020, the root partition is device **md0a**, the swap partition is **md0b**, and the user partition is **md0c**. On an IRIS 3030 the root partition is device **si0a**, the swap partition is **si0b**, and the user partition is **si0f**.

The size of the swap partition determines the maximum amount of virtual memory that can be used. At any moment in time, the sum of the sizes of all the processes known to UNIX cannot exceed the size of the swap partition.

If the simultaneous processes running on your workstation require more space on the swap partition, you can make it larger at the expense of some of the free space in the user file system. You can increase the space available for user files by reducing the swap partition size, but this reduces the total of the process sizes that can exist simultaneously on the system.

To find out the size of the swap partition on your workstation, become the superuser and issue the command:

IRIS 3020:

```
sgilabel md0
```

IRIS 3030:

```
sgilabel si0
```

The *sgilabel* command prints out the disk drive label. The label includes the brand of disk drive and the size of each partition. Table 6-1 also gives the size of each partition for the IRIS 3020/3030.

6.2.1 Changing the Swap Partition Size on an IRIS 3020

To change the size of the swap partition on an IRIS 3020, follow the steps below. For simplicity, the procedure describes a specific, although arbitrarily chosen, change: an increase of 100 cylinders.

1. Log in as root:

```
IRIS login: root
```

2. Back up the */usr* file system and make sure you have a bootable backup tape, in case the root file system is unintentionally corrupted (see Section 4.4).

3. Reboot the workstation:

```
who
sync
reboot
```

4. Boot the disk formatter program:

```
b stand/mdfex
```

CAUTION: You must follow all instructions exactly when you use the *fex* disk formatter programs. Failure to follow instructions could result in serious damage to your file system.

5. When you use the *fex* programs, the monitor doesn't display the letters that you type in, and usually you don't have to press RETURN. Immediately after you type a letter, *fex* responds. When the *mdfex* prompt appears, type:

```
s
```

6. You will see a prompt that says `Set?`. Type the letter *l*:

   ```
   l
   ```

7. You will see this message:

   ```
   File Systems info: lba or cylinder entry?
   ```

   Press `RETURN` twice:

   <return>
   <return>

8. You will see this:

   ```
   a: base: (1)
   ```

   Press `RETURN` to accept this entry.

   <return>

9. Now the IRIS displays additional information:

   ```
   a: base: (1)       size: (150)
   ```

   This entry indicates that partition **md0a** begins at cylinder 1 and has a size of 150 cylinders.

   Partition **md0a** is the root partition and does not change.

10. To add 100 cylinders to the swap partition (**md0b**), you must subtract 100 cylinders from the user partition (**md0c**). The base, or beginning, of partition **c** changes by 100 cylinders. Press `RETURN` to make the IRIS display the base and size entries that are in parentheses.

The new entries are printed in bold below.

Vertex:

```
b: base: (151)        size: (149)249
c: base: (300)400     size: (586)486
```

Maxtor:

```
b: base: (151)        size: (149)249
c: base: (300)400     size: (670)570
```

11. Continue to press RETURN until the workstation displays the `Set?` prompt.

12. Type `q` to quit the `set` function:

```
q
```

13. The workstation displays the `mdfex` prompt. Type `q` again to exit from the *fex* program.

```
q
```

14. You will see this message:

```
Quit
Label on drive 0 needs updating... do it?
```

Give permission to update the drive label:

```
y
```

15. You will see this message:

```
--confirm quit with 'y':
```

Quit from the *fex* program:

```
q
```

16. After a few seconds, the PROM monitor prompt appears. Boot the system in single-user mode:

```
b
```

17. Make the new, smaller user file system:

```
mkfs /dev/rmd0c
```

18. Name the file system and give it a volume name with the *labelit* command:

```
labelit /dev/rmd0c usr sgi
```

19. Mount the */usr* file system:

```
mount /dev/md0c /usr
```

20. Restore the */usr* files from tape:

```
cd /usr
tar xv    or    cpio -ihmud1v
```

21. Return to the root (/) directory and unmount */usr*.

```
umount /dev/md0c
```

22. Check the file system:

```
fsck /dev/rmd0c
```

23. Make a new bootable backup tape (see Section 4.4.1).

24. Begin multi-user mode:

```
multi
```

6.2.2 Changing the Swap Partition Size on an IRIS 3030

To change the size of the swap partition on an IRIS 3030, follow the steps below. For simplicity, the procedure describes a specific, although arbitrarily chosen, change: an increase of 100 cylinders.

1. Log in as root:

```
IRIS login: root
```

2. Back up the */usr* file system on partition **si0f** and make sure you have a bootable backup tape, in case the root file system is unintentionally corrupted (see Section 4.4).

3. Reboot the workstation:

```
who
sync
reboot
```

4. Boot the disk formatter program:

```
b stand/sifex
```

CAUTION: You must follow all instructions exactly when you use the *fex* disk formatter programs. Failure to follow instructions could result in serious damage to your file system.

5. When you use the *fex* programs, the monitor doesn't display the letters that you type in, and usually you don't have to press RETURN. Immediately after you type a letter, *fex* responds. When the `sifex` prompt appears, type:

    ```
    s
    ```

6. You will see the `Set?` prompt. Type the letter *l*:

    ```
    l
    ```

7. You will see this message:

    ```
    File Systems info: lba or cylinder entry?
    ```

    Press RETURN twice:

    <return>
    <return>

8. You will see this:

    ```
    a: base: (1)
    ```

    Press RETURN to accept this entry.

    <return>

9. Now the IRIS displays additional information:

    ```
    a: base: (1)       size: (59)
    ```

    This entry indicates that partition **si0a** begins at cylinder 1 and has a size of 59 cylinders.

    Partition **si0a** is the root partition and does not change.

10. To add 100 cylinders to the swap partition (**si0b**), you must subtract 100 cylinders from the user partition (**si0c**). The base, or beginning, of partition c changes by 100 cylinders. Press RETURN to make the IRIS display the base and size entries that are in parentheses. The new entries are printed in bold below.

    ```
    b: base; (60)       size; (100)200
    c: base; (160)      size; (200)
    b: base; (360)      size; (446)
    c: base; (160)260    size; (646)546
    ```

11. Continue to press RETURN until the workstation displays the `Set?` prompt.

12. Type `q` to quit the `set` function:

    ```
    q
    ```

13. The workstation displays the `sifex` prompt. Type `q` again to exit from the *fex* program.

    ```
    q
    ```

14. You will see this message:

    ```
    Quit
    Label on drive 0 needs updating... do it?
    ```

    Give permission to update the drive label:

    ```
    y
    ```

15. You will see this message:

    ```
    --confirm quit with 'y':
    ```

Quit from the *fex* program:

```
q
```

16. After a few seconds, the PROM monitor prompt appears. Boot the system in single-user mode:

```
b
```

17. Make the new, smaller user file system:

```
mkfs /dev/rsi0c
```

18. Name the file system and give it a volume name with the *labelit* command:

```
labelit /dev/rsi0c usr sgi
```

19. Mount the */usr* file system:

```
mount /dev/si0c /usr
```

20. Restore the */usr* files from tape:

```
cd /usr
tar xv    or    cpio -ihmud1v
```

21. Return to the root (/) directory and unmount */usr*.

```
umount /dev/si0c
```

22. Check the file system:

```
fsck /dev/rsi0c
```

23. Make a new bootable backup tape (see Section 4.4.1).

24. Begin multi-user mode:

```
multi
```

6.2.3 Configuring a Second Disk on an IRIS Workstation

On workstations with two disk drives, you are responsible for configuring the second disk. Table 6-3 summarizes the configuration options.

Configuration	System	Partition	File System Usage
1 File system	**IRIS 3020**	md1g	user files
	IRIS 3030	si1g	user files
3 File systems	**IRIS 3020**	md1a md1b md1c	user files */tmp* user files
	IRIS 3030	si1a si1b si1f	user files */tmp* user files

Table 6-3: Example Configurations of a Second Disk

You can configure your second disk as either one or several file systems. The following two sections give the procedures for configuring a second disk.

Configuring a Second Disk as One File System

To configure a second disk as one large file system, follow these steps:

1. Become the superuser and reboot the system into single-user mode:

```
su
who
sync
reboot
b
```

2. Create a directory in root (/) for the file system to be mounted on **md1g** (for an IRIS 3020) or **si1g** (for an IRIS 3030). Its name must be six characters or less. In this procedure, the name *n* is used.

```
cd /
mkdir n
```

3. Make the user file system:

   IRIS 3020:

```
mkfs /dev/rmd1g
```

   IRIS 3030:

```
mkfs /dev/rsi1g
```

4. Name the file system and give it a volume name with the *labelit* command:

   IRIS 3020:

   ```
   labelit /dev/rmd1g n sgi
   ```

   IRIS 3030:

   ```
   labelit /dev/rsi1g n sgi
   ```

5. Edit */etc/fstab* to contain the following lines:

   IRIS 3020:

   ```
   /dev/md1g   /n   efs   rw,raw=/dev/rmd1g   0  0
   ```

   IRIS 3030:

   ```
   /dev/si1g   /n   efs   rw,raw=/dev/rsi1g   0  0
   ```

   This line automatically mounts and checks the new file system each time multi-user mode is started.

6. Use the *sync* command to flush the new information to the disk.

   ```
   sync
   ```

Configuring a Second Disk as Several File Systems

To configure a second disk as several file systems, follow these steps:

1. Become the super-user and reboot the system in single-user mode:

   ```
   su
   who
   sync
   reboot
   b
   ```

2. Create a directory in root (/) for the file system to be mounted on the disk partitions shown in the table below. The names must be six characters or less. The procedure below uses the names *n1* and *n2*.

System	Partition	File System Name
IRIS 3020	**md1a**	*n1*
	md1b	*tmp*
	md1c	*n2*
IRIS 3030	**si1a**	*n1*
	si1b	*tmp*
	si1f	*n2*

Note that */tmp* already exists and will be mounted on the `b` partition.

```
cd /
mkdir n1 n2
```

3. Make the new file systems:

IRIS 3020:

```
mkfs /dev/rmd1a
mkfs /dev/rmd1b
mkfs /dev/rmd1c
```

IRIS 3030:

```
mkfs /dev/rsi1a
mkfs /dev/rsi1b
mkfs /dev/rsi1f
```

4. Name the file systems and give them volume names with the *labelit* command:

   IRIS 3020:

   ```
   labelit /dev/rmd1a n1 sgi
   labelit /dev/rmd1b tmp sgi
   labelit /dev/rmd1c n2 sgi
   ```

   IRIS 3030:

   ```
   labelit /dev/rsi1a n1 sgi
   labelit /dev/rsi1b tmp sgi
   labelit /dev/rsi1f n2 sgi
   ```

5. Edit */etc/fstab* to contain the following lines:

   IRIS 3020:

   ```
   /dev/md1a   /n1   efs   rw,raw=/dev/rmd1a   0  0
   /dev/md1b   /tmp   efs   rw,raw=/dev/rmd1b   0  0
   /dev/md1c   /n2   efs   rw,raw=/dev/rmd1c   0  0
   ```

   IRIS 3030:

   ```
   /dev/si1a   /n1   efs   rw,raw=/dev/rsi1a   0  0
   /dev/si1b   /tmp   efs   rw,raw=/dev/rsi1b   0  0
   /dev/si1f   /n2   efs   rw,raw=/dev/rsi1f   0  0
   ```

   These lines automatically mount and check the file systems on the second disk each time multi-user mode is started.

6. Enter the *sync* command to flush the new information to the disk.

   ```
   sync
   ```

6.3 IRIS Terminal Disk Configuration

This subsection describes three procedures:

- Changing the swap partition size on an IRIS 3010
- Configuring a second disk on an IRIS 3010
- Creating and using a backup floppy set

6.3.1 Changing the Swap Partition Size on an IRIS Terminal

The disk on the IRIS 3010 is divided into three partitions:

- the root partition (**md0a**), which contains the file system
- the swap partition (**md0b**), which is used by the kernel for paging
- the backup partition (**md0c**), which is a copy of the root partition.

The size of the swap partition determines the maximum amount of virtual memory that can be used. At any moment in time, the sum of the sizes of all the processes known to UNIX cannot exceed the size of the swap partition. To find out the size of the swap partition on your terminal, escape from the *wsiris* terminal emulator with ~!. If you are using the IBM terminal emulator, *t3279*, escape to the shell by following the instructions in *IBM Terminal Emulation*, Escaping to the Shell.

In response to the shell prompt, enter:

```
sgilabel md0
```

The `sgilabel` command prints out the disk drive label. The label includes the brand of disk drive and the size of each partition. Table 6-1 also gives the size of each partition for the IRIS 3010.

If the simultaneous processes running on your IRIS require more space on the swap partition, you can make it larger. If your IRIS 3010 uses less virtual memory than is currently available, you can create more storage room on the disk by reducing the size of the swap partition.

The backup partition is an exact copy of the root partition. To keep the two partitions the same size, divide the space reclaimed from the swap partition equally between the root and backup partitions.

To change the sizes of the partitions, follow these steps:

1. Configure the terminal for PROM monitor boot (i.e., set configuration switches 1 through 5 to `Closed`).

2. Boot the system:

   ```
   b
   ```

3. Escape from the terminal emulator by entering:

   ```
   ~!
   ```

4. Become the super-user:

   ```
   su
   ```

5. Ensure that the backup partition is up to date:

   ```
   rootcopy ok
   ```

6. Mount the backup partition:

   ```
   mount /dev/md0c /mnt
   ```

7. Edit */mnt/etc/inittab* to suppress automatic recovery when booting from the backup partition. Find this line:

   ```
   rr::sysinit:/etc/rootcopy >/dev/console 2>&1
   ```

   Put an `x` between the two colons:

   ```
   rr:x:sysinit:/etc/rootcopy >/dev/console 2>&1
   ```

8. Verify that the backup partition is bootable:

```
reboot
b md0c:recover
```

9. Reboot the terminal into the disk formatter/exerciser, *mdfex*:

```
b stand/mdfex
```

CAUTION: You must follow all instructions exactly when you use the *fex* disk formatter programs. Failure to follow instructions could result in serious damage to your file system.

The system displays the *mdfex* opening screen and prompt.

10. Type `s`. The system displays the *set* prompt:

```
Set ?
```

Respond with the letter `l` to set the disk's label. Press RETURN. The system displays the size of the root partition (partition **a**):

```
a: base: (1)   size: (123)
```

The numbers in parentheses are the current values. The size is expressed in cylinders: each cylinder is 34Kb. Before pressing RETURN, enter the new size for partition **a**. For example, if you are reducing the swap partition by 2 Mb, you will increase the size of partition a by 1 Mb, which represents 30 cylinders.

Respond to the size prompt with the new figure:

```
a: base: (1)   size: (123)153
```

Press RETURN to enter the new size and to display the base cylinder for partition **b**:

```
b: base: ( 124)
```

Change the base for partition **b** to the cylinder immediately following the last cylinder in partition **a**. For example, if you have added 30 cylinders to partition **a**, the new base for partition **b** is 154.

```
b: base: (124)154
```

Press RETURN to enter the new base and to display the size of partition **b**. Enter the new size:

```
b: base: (124)154   size: (360)300
```

11. Continue to press RETURN until the workstation displays the `Set ?` prompt.
12. Type `q` to quit the `set` function.

```
q
```

13. The workstation displays the `mdfex` prompt. Type `q` again to exit from the *fex* program.

```
q
```

14. You will see this message:

```
Quit
Label on drive 0 needs updating... do it?
```

Give permission to update the drive label:

```
y
```

15. You will see this message:

```
--confirm quit with 'y':
```

Quit from the *fex* program:

```
q
```

16. After a few seconds, the PROM monitor prompt appears. Boot the *recover* program, create a filesystem on the newly sized root partition, and copy the backup partition to it:

```
md0c:recover
mkfs /dev/rmd0a
labelit /dev/md0a root sgi
cd /
mount /dev/md0a /mnt
sh # you must use the Bourne shell for the tar command
tar cfB - [!m][!n][!t]* .??* | ( cd /mnt ; tar xvfpB - )
exit
```

    The copy takes about eight minutes.

17. Reboot and change the base and size of the backup partition. Follow the procedure described in step 10, but change the values for partition **c**. Increase the size of partition **c** by the same amount as partition **a**. For the example in step 10, you would enter these figures:

```
c: base: (484)424   size: (123)153
```

    Exit from the *set* routine, update the disk label, and exit from *mdfex*.

18. Boot from the root partition and undo the change to */etc/inittab* made in step 7.

19. Copy the root partition to the backup partition:

```
rootcopy ok
```

6.3.2 Creating and Using a Backup Floppy Set

The IRIS 3010 includes several tools for making a set of floppies for backup and using them to build a file system on the hard disk. The two procedures below describe how to create the floppies and how to build the disk from them.

The floppy set is to be used only in extraordinary circumstances. There is no need for the typical user to create the floppy set; the backup partition on the disk can be used if there is problem with the file system.

Creating the Floppy Set

1. Become the superuser.

   ```
   su
   ```

2. Format eight floppies using *flp*:

   ```
   flp format
   ```

3. Put *mdfex* on one of the formatted floppies:

   ```
   mkfs -G 1 1 /dev/rfloppy 1280
   mount /dev/floppy /mnt
   cp /stand/mdfex /mnt/mdfex
   umount /dev/floppy
   ```

4. Insert a new floppy and issue the *mkbootflp* command to make it a bootable UNIX floppy:

   ```
   mkbootflp
   ```

5. Create a set of *tar* floppies containing the complete system:

   ```
   cd /
   tar cvfb /dev/rfloppy 80
   ```

   Insert a new (formatted) floppy each time the system prompts for a new tape.

Using the Floppy Set

1. Turn on the power or reboot the terminal to enter the PROM monitor.

2. Boot *mdfex* off the *mdfex* floppy:

```
b mf:mdfex
```

CAUTION: You must follow all instructions exactly when you use the *fex* disk formatter programs. Failure to follow instructions could result in serious damage to your file system.

3. Format the hard disk by entering `f`. The system displays a warning and the message:

```
Type 'go <return>' to start ...
```

   Follow the instructions and type:

```
go
```

4. To set the label, enter the letters `s`, then `l`. The system prompts for the name, serial number, and type of drive. Type 3010 after the `Name` prompt. Press RETURN to continue to the next item.

   The serial number is shown on the hard disk drive. Enter the type of drive after the `#?` prompt. Type `21`, the number corresponding to Tandon TM262. Continue to press the RETURN key until the `Set?` prompt appears.

5. Enter `q` twice to quit both the `set` function and `mdfex`. The workstation displays the `mdfex` prompt and the message:

```
Quit
Label on drive 0 needs updating... do it?
```

Confirm your intention to update the drive label by entering `y`. The workstation displays the message:

```
--confirm quit with 'y':
```

7. Type `y` to confirm your intention to quit `mdfex` and return to the PROM monitor. The PROM monitor prompt appears.

8. Boot *defaultboot* from the bootable UNIX floppy:

```
b mf:
```

The system prompts for run level. Enter `s`. Ignore the messages:

```
INIT: cannot open /etc/inittab errno: 2
INIT: execlp of /bin/sh failed; errno = 2
```

9. Transfer the contents of the floppy to partition **a** of the hard disk:

```
/etc/flpuxfer md0a
```

10. Shut down the system and boot *defaultboot* off the hard disk:

```
/etc/reboot
b
```

The system prompts for run level. Answer `s`.

11. Move a few files:

```
/bin/mv /bin/tar /bin/sh /etc/init /tmp
```

12. Insert the first of the six *tar* floppies. Read the complete system onto the disk:

```
cd /
/tmp/tar xvfb /dev/rfloppy 80
```

Insert the subsequent floppies as the system prompts for additional tapes. Insert the floppies in order; otherwise *tar* fails.

13. Copy the system to the backup partition:

```
/etc/rootcopy ok
```

14. Reboot the system:

```
/etc/reboot
```

7. Optional Peripherals

This chapter describes the installation and operation of optional peripherals that you can use with the IRIS. Appendix D has information on cabling and pin signals supported on the IRIS serial ports. Appendix C explains how to install a serial printer on the IRIS. If you purchased an optional color printer, see the document *Using Your Color Printer* that comes with the printer for installation instructions. If you purchased an Apple LaserWriter, see the document *Using Your Laser Printer* for installation instructions.

This chapter covers these topics:

- System console
- ASCII terminal
- Dial and button box
- Modem
- Digitizer tablet
- Tape Drive
- Floppy disk drive
- Stereo optic viewer

The procedures in this chapter assume that your IRIS is booted and that you know how to use the *vi* text editor. If your system is not booted, see Chapter 3; if you do not know how to use a text editor, see *Getting Started with Your IRIS Workstation*.

For reference in the following sections, Figure 7-1 shows the standard I/O panel and auxiliary I/O panel on the back of the IRIS.

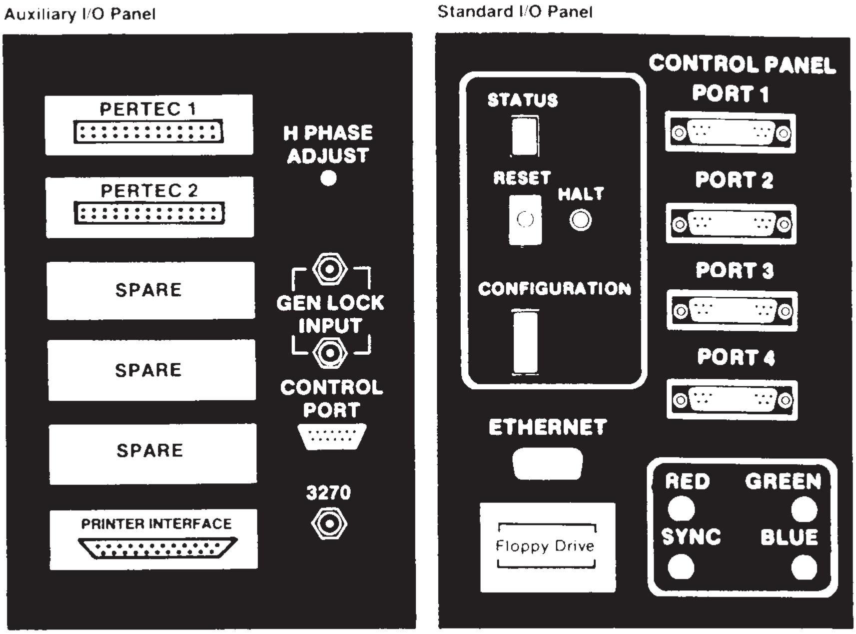


Figure 7-1: Standard I/O Panel and Auxiliary I/O Panel

7.1 Installing the System Console

To install the system console, you can only use `Port 1` of the IRIS workstation. Use the cable supplied with your IRIS to connect the system console to the IRIS.

7.2 Installing the ASCII Terminal

This section describes the procedures for connecting and configuring an ASCII terminal.

7.2.1 Connecting the Hardware

You can connect the ASCII terminal to `Port 2`, `Port 3`, or `Port 4` on the I/O panel. Use a cable that swaps pins 2 and 3 (a null modem cable) and connects pin 7 to pin 7. See Appendix D for cable specifications.

7.2.2 Configuring the Software

To configure UNIX software to use an ASCII terminal with the IRIS, follow these procedures:

1. *IRIS terminal:*

   Escape from the terminal emulator by entering ~! and become the superuser by entering the *su* command.

   IRIS workstation:

   Log in as root or become the superuser by entering the *su* command.

2. Look at the /etc/termcap file by typing:

   ```
   more /etc/termcap
   ```

   For most ASCII terminal models, you do not need to edit the file */etc/termcap*. The */etc/termcap* file describes different terminal models, their capabilities, and how they operate. Look at the */etc/termcap* file to see if your terminal model is in this file. If your terminal model is not in this file, or if your terminal does not work properly after you set up the software configurations, put an entry in this file for your terminal model. Refer to the manual pages for *termcap*(4), *tset*(1), *ttytype*(4), and *stty*(1) in the *UNIX Programmer's Manual, Volume IA and IB*.

For example, if your terminal is a Visual 50, the entry looks like this:

```
vj|v50am|v50 with automatic margins:\
```

The data in the second field (v50am) corresponds to the model of your terminal. You need to know how the model description appears in this field so that you can enter it in the file */etc/ttytype* in the next step.

3. Edit the file */etc/ttytype* by typing:

```
vi /etc/ttytype
```

The file */etc/ttytype* defines the type of terminal being used and the port to which it is connected.

The */etc/ttytype* file looks like this:

```
wsiris      systty
wsiris      console
wsiris      syscon
?du         ttyd1
?du         ttyd2
?du         ttyd3
```

In the line that corresponds to the port you are using, replace the du with the terminal model name from */etc/termcap*. `ttyd1` is `Port 2`, `ttyd2` is `Port 3`, and `ttyd3` is `Port 4`.

For example, to connect a Visual 50 to `Port 2`, change */etc/ttytype* to look like this:

```
wsiris      systty
wsiris      console
wsiris      syscon
?v50am      ttyd1
?du         ttyd2
?du         ttyd3
```

The `?` causes *tset* to prompt for the kind of terminal you are using when you log on through that port. You normally call *tset* in your login startup script (*~/.cshrc* or *~/.profile*). *tset* commands use information from */etc/ttytype* and */etc/termcap* to initialize the terminal. These files also provide information on setting up environment variables so that editors and other programs know how to communicate with the terminal. See Chapter 4 and the *tset*(1) manual page.

4. Edit the file */etc/inittab* to enable you to log in to the ports of the IRIS. (See the *inittab*(4) manual page for more information.) This is an example */etc/inittab* file:

```
is:s:initdefault:
s0::sysinit:/etc/rc.s0 1>/dev/console 2>&1
b0::bootwait:/etc/brc </dev/console >/dev/console 2>&1
b1::bootwait:/etc/bcheckrc </dev/console >/dev/console 2>&1
rc::wait:/etc/rc 1>/dev/console 2>&1
pf::powerfail:/etc/powerfail 1>/dev/console 2>&1
co::respawn:/etc/getty console co_9600 none LDISC0
d1:x:respawn:/etc/getty ttyd1 dx_9600 none LDISC0
d2:x:respawn:/etc/getty ttyd2 dx_9600 none LDISC0
d3:x:respawn:/etc/getty ttyd3 dx_9600 none LDISC0
```

Here is an example of one entry, with an explanation of each field in the entry:

```
d1:x:respawn:/etc/getty ttyd1 dx_9600 none LDISC0
```

`d1` uniquely identifies the entry.

`x` defines the run level in which this entry is to be processed.

An `x` in this field means "never do this"; nothing in this field means "always do this"; an `s` means "do this when switched to single-user mode"; a `2` means "do this in run-level 2"; a `3` means "do this in run-level 3"; a `23` means "do this in run-level 2 or 3".

`respawn` defines the action on the process field of *init*. Refer to the manual page *inittab*(4) in the *UNIX Programmer's Manual, Volume IB*, for a description of all the actions of the process field of *init*.

```
/etc/getty ttyd1 dx_9600 none LDISC0
```

runs the `getty` process on `Port 2` at the baud rate and options specified in the `dx_9600` entry in the file */etc/gettydefs*.

To enable you to log in to the terminal you connected to `Port 2`, remove the `x` in the second field of the entry for `Port 2` and add a `23`. Find this line:

```
d1:x:respawn:/etc/getty ttyd1 dx_9600 none LDISC0
```

Edit it so that it looks like this:

```
d1:23:respawn:/etc/getty ttyd1 dx_9600 none LDISC0
```

The file */etc/inittab* refers to the file */etc/gettydefs* for information concerning the terminal baud rate. If the terminal will be run at a baud rate other than 9600, replace `dx_9600` with the name of the entry in */etc/gettydefs* that has the correct baud rate. The speeds available at the start of the getty program can be examined or changed in the */etc/gettydefs* file. The entries in */etc/gettydefs* look like this:

```
dx_9600# B9600 # B9600 SANE TAB3 #\r\n\n$HOSTNAME login: #dx_9600
dx_4800# B4800 # B4800 SANE TAB3 #\r\n\n$HOSTNAME login: #dx_4800
dx_2400# B2400 # B2400 SANE TAB3 #\r\n\n$HOSTNAME login: #dx_2400
dx_1200# B1200 # B1200 SANE TAB3 #\r\n\n$HOSTNAME login: #dx_1200
du_1200# B1200 # B1200 SANE TAB3 #\r\n\n$HOSTNAME login: #du_1200
du_300#  B300  # B300  SANE TAB3 #\r\n\n$HOSTNAME login: #du_300
du_2400# B2400 # B2400 SANE TAB3 #\r\n\n$HOSTNAME login: #du_1200
```

The entries beginning with `dx` are for terminals; those beginning with `du` are for modems. See the manual page *gettydefs*(4) for more information on the fields of each entry.

When the terminal is powered on, the workstation sends a login prompt to the terminal screen. Press Return if the login prompt doesn't appear

If the default line speed set in */etc/inittab* is incorrect, the prompt can be garbled or may not appear.

5. Inform *init* of the change to */etc/inittab* and start a *getty* process for the port:

```
telinit q
```

7.3 Installing the Dial and Button Box

The dial and button box has 8 continuous rotation dials, 32 function switches, 32 programmable indicator lights, and an 8-character display.

7.3.1 Installing the Hardware

To install the dial and button box, follow these procedures:

1. Connect the dial box to the button box with the 37-pin cable assembly supplied.

2. Find the cable with a DB9 connector at one end and a DB25 connector at the other end. Attach the DB9 connector to the button box and the DB25 connector to Port 2, Port 3, or Port 4 of the IRIS workstation.

3. Plug the 3-wire power cord from the button box into a 3-wire AC power outlet.

7.3.2 Installing the Software

To install the dial and button box software, follow these procedures:

1. *IRIS terminal:*

   Escape from the *wsiris* terminal emulator by entering ~! and become the superuser by entering the *su* command.

IRIS workstation:

Log in as `root` or become the superuser by entering the *su* command.

2. Edit the file */etc/inittab* so you cannot log in to the port to which the dial and button box is connected. For example, if you connected the dial and button box to `Port 3`, find the line for *ttyd2*, and make sure it has an `x` in the second field:

   ```
   d2:x:respawn:/etc/getty ttyd2 dx_9600 none LDISC0
   ```

   The `x` in the second field doesn't let you log in by preventing a *getty* process from being run for the port.

3. Inform *init* of the changes to */etc/inittab* by typing:

   ```
   telinit q
   ```

   This command instructs *init* to read */etc/inittab* and kill the *getty* process for the port.

4. Issue the *devport* command to connect the dial and button box to `Port 3`:

   ```
   devport -d 2
   ```

5. Edit the file */etc/rc.local*. Add the *devport* command shown in the previous step. This ensures that the port for the dial and button box is specified automatically each time you boot the IRIS.

6. Give all users permission to access the port:

   ```
   chmod 666 /etc/ttyd2
   ```

7. Change the owner of the device to `root` and its group to *sys*:

```
chown root /dev/ttyd2
chgrp sys /dev/ttyd2
```

To use the dial and button box and the digitizer tablet simultaneously, see *devport*(1) in the *UNIX Programmer's Manual, Volume IA*, and `devport`(3G) in the *IRIS User's Guide*.

The Graphics Library routine `dbtext` sets the dial and button box text. See the *IRIS User's Guide* for more information.

7.4 Installing the Modem (IRIS Workstation Only)

This section describes the procedures for installing the modem hardware, configuring a port for a dial-in modem and configuring a port for a dial-out modem.

7.4.1 Installing the Hardware

The IRIS workstation supports all 110 through 2400 baud modems. The Bell 212, 103, and 113 models; the VADIC 3400; the CCITT Recommendation V.25 and V.22, and the Hayes Smartmodem are all examples of modems that the IRIS supports. Connect any of these modems to either `Port 2`, `Port 3`, or `Port 4` of the IRIS with an RS-232 cable.

Only some of the ports support *data carrier detect* and *data terminal ready*. See the tables in Appendix D for information on the IRIS port signals.

7.4.2 Configuring the Software for a Dial-out Modem

This section describes the configuration procedures for a dial-out modem. Refer to "*uucp* Administration", in the *UNIX Programmer's Manual, Volume IIB*.

The following example shows you how to connect and configure the software for a modem.

1. Log in as `root` or become the superuser by entering the *su* command.

2. Edit the file */usr/lib/uucp/L-devices*. This file contains line speed entries for each port.

   The first field of this entry defines the connection as direct (DIR).

   The second field is the device name. *ttym1* corresponds to `Port 2`, *ttym2* corresponds to `Port 3`, and *ttym3* corresponds to `Port 4`.

   The third field is a place holder.

   The fourth field is the baud rate of the connection. Use a baud rate that works for both the local modem and the modem on the other end of the telephone line.

   Add this line for *ttym1* if you are connecting the modem to `Port 2`:

   ```
   DIR ttym1 ttym1 baudrate
   ```

   For example, if your modem is operating at 2400 baud, add this line:

   ```
   DIR ttym1 ttym1 2400
   ```

   Add this line for *ttym2* if you are connecting the modem to `Port 3`:

   ```
   DIR ttym2 ttym2 baudrate
   ```

   Add this line for *ttym3* if you are connecting the modem to `Port 4`:

   ```
   DIR ttym3 ttym3 baudrate
   ```

3. Edit the file */usr/lib/uucp/L.sys*. This file has information about sites that *uucp* can communicate with. Entries in this file have these fields:

   system_name time device class phone login

system_name	the name of the remote system; must be unique in the first seven characters.
time	when the remote system can be accessed; ANY means any day of the week.
device	the device used for the call.
class	the baud rate for the device.
phone	the dialer sequence the modem will use to call the remote system.
login	the login password and special character sequence needed to complete the login.

   For example, if you are attaching a 1200-baud modem to `Port 4`, and you want it to automatically call a host named *omachine* at 415-555-1212, you might enter these lines in the */usr/lib/uucp/L.sys* file:

```
# Name of company
#talk to (415)555-1212, Tom Jones.
#mail to omachine!him
#incoming=Upyramid, password cube
omachine Any ttyd3 1200 ttyd3 "" \rATS2=128\r\c
OK atdt4155551212\r 1200
\r\c ogin: -\b\d-ogin:--ogin:--ogin:
Uname ssword: PaSsWoRd
```

   NOTE: The line that begins with omachine and ends with PaSsWoRd is actually one line. Do *not* type any carriage returns. UNIX will wrap around as the line reaches the edge of the screen.

4. To configure the port for the modem, type these commands:

   ```
   cd /dev
   ./MAKEDEV duart
   ```

   This step automatically makes a directory entry for the modem, gives users permission to access the port, and changes the owner of the device to `root` and its group to *sys*.

5. Edit the file */etc/inittab* to disable logins on the port connected to the modem. Find the line that corresponds to the port you chose for the modem. *ttyd1* corresponds to `Port 2`, *ttyd2* corresponds to `Port 3`, and *ttyd3* corresponds to `Port 4`. For example, if you connected the Hayes modem to `Port 3`, find this line:

   ```
   d2:x:respawn:/etc/getty ttyd2 dx_9600 none LDISC0
   ```

   and make sure it has an `x` in the second field. The `x` in the second field doesn't let you log in by preventing a *getty* process from being run for the port.

6. Inform *init* of the change to */etc/inittab* by typing the command:

   ```
   telinit q
   ```

7. Test the serial line with *cu*:

   `cu -s`*baudrate* `-l`*device*

   The IRIS should connect to the direct line specified by the **–l** option. See the manual page *cu*(1C) in the *UNIX Programmer's Manual, Volume IA*, for more information.

7.4.3 Configuring the Software for a Dial-in Modem

This section describes the configuration procedures for a dial-in modem. Refer to "*uucp* Administration", in the *UNIX Programmer's Manual, Volume IIB*.

1. Log in as `root` or become the superuser by entering the *su* command.

2. To configure the port for the modem, type these commands:

```
cd /dev
./MAKEDEV duart
```

   This step automatically makes a directory entry for the modem, gives users permission to access the port, and changes the owner of the device to `root` and its group to *sys*.

3. Edit the file */etc/inittab* to disable logins on the port connected to the modem, and to specify that the modem device driver be used with that port. Find the line that corresponds to the port you chose for the modem. *ttyd1* corresponds to `Port 2`, *ttyd2* corresponds to `Port 3`, and *ttyd3* corresponds to `Port 4`. Edit the line, changing `d1` to `m1`, `d2` to `m2`, or `d3` to `m3`. For example, if you connected the Hayes modem to `Port 3`, find this line:

```
d2:x:respawn:/etc/getty ttyd2 dx_9600 none LDISC0
```

   and modify it to read:

```
d2:23:respawn:/etc/getty ttym2 du_1200 none LDISC0
```

   Make sure there is a `23` in the second field so you can log in to the port.

4. Inform *init* of the change to */etc/inittab* by typing the command:

```
telinit q
```

The du entries in */etc/gettydefs* toggle between 1200 and 300 baud. If *getty* is listening at the wrong baud rate when you log in, the prompt can be garbled or may not appear. Select another line speed by pressing the `BREAK` key after the login message is displayed. See the manual pages *gettydefs*(4) and *getty*(1M).

7.5 Installing the Digitizer Tablet

The IRIS supports the Hitachi digitizer tablet, which converts graphics information into digital form. The resolution of the digitizer is 200 points per inch. The tablet has a serial interface and is shipped configured for 9600 baud communication and output in binary format.

7.5.1 Installing the Hardware

To install the digitizer tablet, follow these procedures:

1. Locate the dip switches labeled `DSW1`, `DSW2`, and `DSW3` on the back of the digitizer tablet. The tablet is shipped with the switches properly set. Up means `On` and down means `Off`. Make sure that the switches match the settings in the following list:

DSW1	Status	Meaning
1	On	Unit selection - English (inch)
2	Off	Mode selection - run
3	Off	Mode selection - run
4	Off	Incremental mode - normal
5	Off	Rate selection - 1/20
6	Off	Rate selection - 1/20
7	Off	Reset - normal
8	Off	Not used

DSW2	Status	Meaning
1	Off	Baud rate - 9600
2	Off	Baud rate - 9600
3	On	Baud rate - 9600
4	On	Not used
5	Off	Serial data transfer - 2 stop bits
6	On	Word composition - odd parity
7	On	Bit pad compatible modem
8	Off	Digitize/test - digitize

DSW3	Status	Meaning
1	On	ASCII/binary
2	Off	Not used
3	Off	Cursor button code - 4-button cursor
4	Off	Not used
5	Off	ASCII format selected - L/F code attached
6	On	Buzzer disable
7	Off	Resolution - 200 points per inch
8	Off	Resolution - 200 points per inch

2. The *tablet cable* is a null modem cable that also joins pins 4, 5, 6, 20, and 8 at the tablet end. Attach the end labeled `tablet` to the port on the digitizer tablet labeled `PORT`. Connect the other end to the IRIS I/O panel. You can use `Port 2`, `Port 3`, or `Port 4`. This example uses `Port 4`.

3. Connect the four button cursor to the port labeled `STYLUS` on the back of the tablet. Plug the tablet power cable into an AC outlet.

7.5.2 Installing the Software

To install the digitizer tablet software, follow these procedures:

1. *IRIS terminal:*

   Escape from the *wsiris* terminal emulator by entering ~! and become the superuser by entering the *su* command.

   IRIS workstation:

   Log in as `root` or become the superuser by entering the *su* command.

2. Edit the file */etc/inittab* to disable logins on the port you chose for the digitizer tablet. This example uses `Port 4`, device *ttyd3*. Find the line for *ttyd3*, and make sure it has an `x` in the second field:

   ```
   d3:x:respawn:/etc/getty ttyd3 dx_9600 none LDISC0
   ```

   The `x` in the second field doesn't let you log in by preventing a *getty* process from being run for the port.

3. Tell *init* of the changes to */etc/inittab* by typing:

   ```
   telinit q
   ```

   This command informs *init* to read */etc/inittab* and kill the *getty* process for the port.

4. Enter the *devport* command to assign a port to the digitizer tablet:

   ```
   devport -d 3
   ```

5. Edit the file */etc/rc.local*. Add the *devport* command shown in the previous step. This ensures that the port for the digitizer tablet is specified automatically each time you boot the IRIS.

6. Give all users permission to access the port:

   ```
   chmod 666 /etc/ttyd3
   ```

7. Change the owner of the device to `root` and its group to *sys*:

```
chown root /dev/ttyd2
chgrp sys /dev/ttyd2
```

7.6 Installing the Tape Drive (IRIS Workstation Only)

The sections below describe how to install both half-inch and quarter-inch tape drives on the IRIS workstation.

7.6.1 Half-inch Tape Drive

The half-inch PE tape unit writes data in PE (Phase Encoded) mode at 1600 bpi (bits per inch). Data is written on the tape by positively or negatively magnetizing sections of the oxide on the tape. A change in direction from positive to negative represents 1 bit and a change in direction from negative to positive represents 0 bit.

Connecting the Stand-alone Tape Drive to the Workstation

The half-inch tape drive is a stand-alone unit. The tape drive is connected to the electronics cabinet on the auxiliary I/O panel with two 50-pin ribbon cables. Figure 7-1 shows the two connectors on the auxiliary I/O panel, labeled `Pertec 1` and `Pertec 2`, for connecting the half-inch tape drive to the electronics cabinet. For instructions on connecting the tape unit to the electronics cabinet, see the *PE Tape Unit Operator's Guide* that came with your optional tape drive.

Using the Tape Drive with the Workstation

You can use the tape drive for backup up the disk, but you cannot boot from the half-inch tape unit.

For information on the operator panel, loading tapes, error conditions, and cleaning the tape unit, see the *PE Tape Unit Operator's Guide*.

See section 4.4 for instructions on backing up the disk to the tape drive. The half-inch tape device with rewind is device */dev/rmt3*; the half-inch tape drive without rewind is device */dev/rmt4*.

7.6.2 Quarter-inch Cartridge Tape Drive

The quarter-inch cartridge tape drive is installed in the workstation at the factory if the workstation is ordered with this option. The specifications for the quarter-inch tape drive are shown in Table 7-1.

Cartridge Tape Drive Specifications	
Device names:	/dev/rmt1 (with rewind)
	/dev/rmt2 (without rewind)
Density:	10,000 flux changes per inch
	100K per foot
Speed:	90 inches per second
Tape Lengths:	450 and 600 feet
Capacity (450 ft.)	40 MB
(600 ft.)	60 MB
Tape Suppliers:	3M/Scotch
	Data Electronics, Inc.
	Control Data Corporation

Table 7-1: Cartridge Tape Drive Specifications

7.7 Using the Floppy Disk Drive

The floppy disk drive is installed at the factory if the IRIS is ordered with this option. It is standard on the IRIS terminal. The floppy disk drive is device */dev/floppy*.

The floppy disk drive has an unformatted capacity of 1 Mb per disk. It has a data transfer rate of 250K bits per second, 96 tracks per inch, and 160 tracks per drive.

To create and use a set of backup floppies for IRIS terminals, see Section 6.3.

7.8 Installing the Stereo Optic Viewer

Silicon Graphics supports a stereo viewer manufactured by Stereo-Optic Systems, Inc. (model number SDS-100). The stereo viewer is not available from Silicon Graphics.

7.8.1 Connecting the Stereo Optic Viewer to the IRIS

To connect the stereo viewer with the IRIS, use the nine-pin D to BNC adapter cable provided by Silicon Graphics (product number C-STV).

1. Connect the BNC connector to the port labeled `TTL Trigger Input` on the stereo viewer back panel.
2. Connect the nine-pin D connector to the port labeled `Control Port` on the auxiliary I/O panel on the IRIS (see Figure 7-1).

7.8.2 Using the Stereo Optic Viewer

The IRIS sends a pulse through the cable each time the color map is cycled or the front and back buffers are swapped. If both cycling and swapping are taking place, the pulses follow the cycle rate and ignore the swap rate. (For more information, see the descriptions of `cyclemap` and `swapbuffers` commands in the *IRIS User's Guide*.)

8. Video Options

The IRIS workstation supports seven video options:

- 60-Hz non-interlaced (fifteen-inch or nineteen-inch)
- 33-Hz interlaced monitor
- RS-170A video (genlockable and non-genlockable)
- European video standard (genlockable and non-genlockable)

NOTE: Early IRIS series 2000 models with the Turbo upgrade may not have the same monitors described here.

Each IRIS workstation contains drivers for two video options, one *primary* video option and one *secondary* video option. Table 8-1 lists the possible combinations of primary and secondary video options.

Primary Video	Secondary Video
60-Hz Non-interlaced	33-Hz Interlaced
60-Hz Non-interlaced	RS-170A
60-Hz Non-interlaced	European Video Standard
33-Hz Interlaced	60-Hz Non-interlaced
33-Hz Interlaced	RS-170A
33-Hz Interlaced	European Video Standard

Table 8-1: Primary and Secondary Video Options

NOTE: The RS-170A and European video standard monitors are not available from Silicon Graphics.

The configuration for the primary and secondary video drivers is specified when an IRIS workstation is ordered. See below for the procedures for using the IRIS Graphics Library with different video options.

8.1 Supported Options

The sections below describe the video options supported by the IRIS workstation.

8.1.1 60-Hz Non-interlaced Monitor

The 60-Hz non-interlaced high-resolution monitor has a visible resolution of 1024 pixels by 768 lines. These monitors are available in two sizes: fifteen-inch and nineteen-inch. For more information on these monitors, see Chapter 2.

8.1.2 33-Hz Interlaced Monitor

The 33-Hz interlaced high-resolution monitor has a visible resolution of 1024 pixels by 768 lines. For more information, see Chapter 2.

8.1.3 RS-170A

The RS-170A option has a visible resolution of 636 pixels by 485 lines and a frame rate of 30 Hz.

The IRIS workstation provides red, green, blue, and sync video outputs (RGBS). The RGB outputs from the IRIS are noncomposite 0.7 V p-p into 75 ohms, positive bright. The sync output is TTL, low active, and is capable of driving a 75-ohm load. It meets RS-170A timing with one exception: the oscillator that determines horizontal scanning frequency stability in the IRIS differs from the frequency stability of the RS-170A color television timing standard (± 0.005% versus ± 0.000279%). This stability is adequate tor most applications but not for broadcasting. For broadcast applications, use either the genlockable RS-170A option or a timebase corrector.

The IRIS RGBS output can be encoded into an NTSC signal with a stand-alone NTSC color encoder or an NTSC color encoder connected to a sync generator genlocked to the IRIS sync output. This NTSC signal can be connected to video tape recorders or an NTSC-type monitor.

8.1.4 European Video Standard

The European video standard has a visible resolution of 780 pixels by 575 lines, a frame rate of 25 Hz, and a total vertical resolution of 625 lines.

The IRIS workstation provides red, green, blue, and sync video outputs (RGBS). The RGB outputs from the IRIS are noncomposite 0.7 V p-p into 75 ohms, positive bright. The sync output is TTL, low active, and is capable of driving a 75-ohm load. It meets European low-resolution video timing, with one exception: the oscillator that determines horizontal scanning frequency stability in the IRIS differs from the frequency stability of the European color television timing standards (± 0.005% versus ± 0.0001%). This stability option is adequate for most applications but not for broadcasting. For broadcast applications, use the genlockable European low-resolution video or a timebase corrector.

The IRIS RGBS output can be encoded into a PAL signal by a stand-alone PAL color encoder or a PAL color encoder connected to a sync generator genlocked to the IRIS sync output. This PAL signal can be connected to a video tape recorder or a PAL-type monitor.

8.1.5 Genlockable RS-170A or European Video Standard

The genlockable RS-170A or European video option synchronizes the IRIS RGB outputs with those of an external sync master.

The IRIS RGBS output can be encoded into an NTSC or PAL signal with a stand-alone color encoder or a PAL color encoder connected to a genlockable sync generator which is also genlocked to the sync master.

The genlockable RS-170A or European video option provides operation with commercially available video tape recorders and mixing/keying equipment (when color encoded with external equipment). If a genlockable video option is ordered, two BNC connectors are provided with the IRIS workstation system. These connectors are located on the auxiliary I/O panel and are labeled `Genlock Input` (see Figure 7-1). If the IRIS is not used in a daisy chain, a 75-ohm terminator is required on one of the two BNC connectors.

Operation of the genlockable RS-170A or genlockable European Video Standard options are identical. The IRIS must be operating in its supported low-resolution mode (RS-170A or European Video Standard) to genlock. Genlock provides for vertical and horizontal alignment of the IRIS RGB and Sync outputs to an external composite sync reference input presented to the `Genlock Input` on the auxiliary I/O panel (see Figure 7-1). This input is a loop-through connection requiring external 75-ohm termination.

The genlocking process takes approximately two seconds. When a genlock reference input is connected, the genlocking process is initiated automatically. If no composite sync reference input is connected, the appropriate output (RS-170A or European Video Standard) runs referenced to an internal timing source. Genlocking takes place whenever the IRIS is reset, the genlock reference input is disconnected and reconnected, or an out-of-lock condition is detected. An out-of-lock condition typically results from a discontinuity on the genlock reference input, such as a power interruption or a source switch.

When genlocked, the IRIS output may be adjusted horizontally to compensate for external equipment delays. To do this, use the horizontal phase adjust potentiometer, labeled `HPhase Adjust`, which is reached through an access hole on the auxiliary I/O panel (see Figure 7-1). This potentiometer provides an adjustment range of ± 1.5 microseconds on the IRIS RGBS output with respect to the genlock reference input.

8.2 Booting with the Secondary Video Driver

Configuration switch 7 on the cabinet I/O panel controls which video driver is initialized when the workstation is booted in PROM monitor boot configuration.

To boot using the secondary display:

1. Set switch 7 to `Open` to initialize the video hardware to drive the secondary display. (Setting switch 7 to `Closed` causes the PROM monitor to initialize the hardware to drive the primary display.)

2. Set the `Power` switch on the cabinet front panel to `On`.

   The PROM monitor prompt appears on the screen:

   ```
   iris>
   ```

   If the prompt does not appear, press the `Reset` button.

3. Boot UNIX. To boot from the default device, type:

   ```
   b
   ```

   See Table 3-2 for a list of PROM monitor commands.

Instead of booting using the PROM monitor, the secondary display can be selected using the `setmonitor` command (see below).

NOTE: Although it can be used as the secondary display, the RS-170A monitor option cannot be used as the boot display.

8.3 Using the Graphics Library with the Video Options

The setmonitor command in the IRIS Graphics Library specifies the video compatibility of the IRIS workstation display output. To change the active video driver through the Graphics Library, issue the command:

```
setmonitor(type)
```

Specify one of these types: HZ60 (60-Hz non-interlaced monitor), HZ30 (33-Hz interlaced monitor), NTSC (RS-170A monitor), or PAL (European video standard).

These monitor types are defined from the include file *set.h*. The command `getmonitor`, which returns the current display controller mode, returns one of these types. (For more information, see the *IRIS User's Guide*.)

With a low-resolution monitor, you should also reduce the size of the viewport, using the `viewport` command:

```
viewport(left,right,bottom,top)
```

To map the graphics output to the entire screen, specify the exact available pixel range:

NTSC monitor

```
viewport(0,635,0,484)
```

PAL (European video standard) monitor

```
viewport(0,779,0,574)
```

The `ortho` commands control the mapping of graphics coordinates to screen coordinates. Used in conjunction with the `viewport` command, the `ortho2` command lets you program in screen coordinates on low-resolution monitors. To establish a one-to-one mapping of graphics coordinates to screen coordinates, use parallel ranges in the two commands:

NTSC monitor

```
ortho2(-0.5,635.5,-0.5,484.5)
```

PAL monitor

```
ortho2(-0.5,779.5,-0.5,574.5)
```

If the graphics programs are run on a low-resolution monitor with the default viewport specifications, only the lower left portion of the image will be visible on the screen. UNIX will not work in NTSC or PAL mode because it requires more pixel area. To run UNIX, switch to a high-resolution monitor with `setmonitor`.

If you design a program to run on both high-resolution and low-resolution monitors, keep in mind that fonts have a fixed pixel width. This means that text on one monitor will appear different on another monitor.

9. Demonstration Programs and Gifts

The IRIS workstation is shipped with a set of demonstration programs, programming examples, and other miscellaneous tools. This chapter tells how to operate the demonstration programs. The demonstration programs that run under the window manager are described in detail in Section 9.1.3.

9.1 Demonstration Programs

Demonstration programs can be run any time after the system is booted. They are located in the directories */usr/people/demos* and */usr/people/mexdemos*. The directory *mexdemos* contains demonstration programs that run with the window manager; the programs in *demos* do not work correctly if the window manager is running .

9.1.1 Running Demonstration Programs

To run demonstration programs, follow these procedures:

1. Boot the workstation in multi-user mode.

2. Log in to the workstation as *demos* or *mexdemos*.

```
IRIS login: demos
IRIS login: mexdemos
```

3. If you are running the window manager demonstration programs (*mexdemos*), run *startup*:

```
startup
```

The textport is labeled with a banner, which indicates it is the console window.

4. Display a listing of the available programs with the *ls* command:

```
ls
```

5. Enter the name of the program you want to run. For example, to use the flight simulator, enter the word *flight*, followed by RETURN.

```
flight
```

Some of the demonstration programs exit automatically when they finish running. To exit the open-ended demonstration programs or to interrupt a program, press ESC. To clear the screen of graphics "leftovers" if you are not running the window manager demonstrations, enter the command *gclear*(1G).

More information on the demonstration programs is included in Section 1 of the on-line *UNIX Programmer's Manual*. Manual pages for the window manager commands are labeled 1W; other demos are labeled 1D.

9.1.2 Using the Mouse with the Window Manager

There are three buttons on the mouse. They are named LEFTMOUSE, MIDDLEMOUSE, and RIGHTMOUSE. Almost all the window manager functions can be controlled by the mouse x-y valuators and the RIGHTMOUSE.

Before continuing, make sure you understand how to use the mouse to communicate with windows. This information is located in the *IRIS User's Guide*.

9.1.3 Window Manager Demonstration Programs

The demonstration programs described in this section fall into three main groups.

- *2D demonstration programs* illustrate how the window manager works and provide instruction on the color-handling capabilities of the IRIS.
- *3D object manipulation* demonstration programs include *arch*, *flow*, *heme*, *jet*, and *shuttle*. These programs move 3D objects around. The IRIS can perform simultaneous manipulation of different 3D objects in independent windows.
- *Shaded object* and *curved surface* demonstration programs illustrate shading and surface treatment on the IRIS.

Window manager demonstration programs require a large number of bitplanes. *flow* requires 12; *arch*, *heme*, *jet*, *shuttle*, and surface editors require 24; *zshade* requires 32.

2D Window Manager Demonstration Programs

A set of 2D programs is included in the collection of demonstration programs. These tools include the following programs: *bckgrnd*, *showmap*, *demomakemap*, *showramp*, *gamma*, *cedit*, *interp*, and *mag*. These tools can be classified into three categories: backgrounds, color manipulation programs, and pixel access programs. To start any of these tools, type the name of the tool in the console window.

Backgrounds

bckgrnd is a non-interactive, textured wallpaper with art deco designs. A particular background fills all available space which is not covered with another window.

Color Manipulation Programs

showmap is a window filled with up to 1024 little squares. It displays the current colors in the color map, counting from the color 0 at the lower left corner up to the largest color index at the top right. (For more information on color maps, see the *IRIS User's Guide* and the *makemap*(1W) command in the *UNIX Programmer's Manual*.)

demomakemap makes and remakes the color map for the demonstration programs. Before other graphics programs are executed, run *demomakemap* to build the default color map for the demonstration package. To see the effect of changing the color map using *demomakemap*, run *showmap* in a separate window. (At any time during the demonstration session, the original color map can be restored by executing *demomakemap*.)

demomakemap also works in conjunction with the *gamma* program. *demomakemap* reads the value from the file *.gamma* in your home directory and uses it to calculate its shade scales (see below).

showramp is a window with three Gouraud-shaded rectangles. The rectangles are colored smoothly using physical, chromatical adjacent color indices that form a color ramp in the color map.

The *gamma* program uses a technique called gamma-correction to change the smoothness of the transitions between the color intensities in a color map. The physical characteristics of the CRT phosphor, the ambient lighting, and the vision and preference of the viewer affect the perception of the color map. A strictly linear color ramp often appears too dark, especially towards the middle of the ramp. *gamma* allows the increments from color to color to be exponential, and thus, the middle of the color ramp can be brighter or darker than in a linear color ramp.

The *gamma* command accepts a single floating-point argument. A value of 1.0 creates a strictly linear color ramp. Values above 1.0 increase the brightness in the middle of a color ramp; values below 1.0 decrease the brightness. Values between 1.0 and 3.0 are typical.

The *gamma* program does not immediately affect the color map. It stores its argument in a file called *.gamma* in your home directory. Whenever a color ramp is made, such as in the program *demomakemap*, this file is referenced for a value for *gamma*. The *gamma* program itself does not open a window.

Below is an exercise in running the previously described programs. To run these programs, use the following procedure:

1. Bring up the windows *showmap* and *showramp*.

2. Set the gamma correction value and use it to change the color map:

   ```
   gamma 0.5
   demomakemap
   ```

   Wait a moment for the colors to change. Observe the effects on the object and the brightness and smoothness of the color ramps in the *showmap* and *showramp* windows. (*demomakemap* does not open a window.)

3. Repeat the experiment with a different *gamma* value:

   ```
   gamma 3.0
   demomakemap
   ```

   Remember that the value you select for *gamma* is saved until you change it.

cedit is an interactive color editor in a graphics window. To use *cedit*:

1. Bring up the *cedit* window and select it for input.

2. Move the cursor to a pixel outside the *cedit* window itself, to select a color index for editing.

3. To choose the color index of the pixel, press the LEFTMOUSE button and release it.

   The selected color index is painted on the sample chip on the right of the *cedit* window, and the color sliders are set to the red, green, and blue values of that color.

4. To change the intensity of the color value of the sample chip, press the LEFTMOUSE button down while the cursor is located on one of the three color sliders and move the sliders up and down.

   Try picking a color from *showmap* and editing it with *cedit*.

interp displays and alters a region of the color map. Use *interp* to make color ramps for depth-cued or smooth-shaded objects. *interp* makes a color ramp of any size between any two color indices, interpolating the red, green and blue intensities of all indices between the two prescribed extremes. Two squares show the extreme colors of a color ramp and a long rectangle shows the intermediate color range.

The following sequence describes how to use *interp* with *showmap* and *cedit* to create a user-defined color ramp.

1. Attach to the *cedit* window. Use *cedit* to edit any two color indices displayed by *showmap*.

2. Detach from *cedit* and attach to *interp*. Point to one of the extreme colors from *showmap* that you have edited and press the LEFTMOUSE button.

4. Point to the other extreme color and again press the LEFTMOUSE button. The colors appear as two large squares in the *interp* window. (If you make a mistake while choosing extreme colors, for example, the cursor is over an incorrect color index, then continue picking colors with the LEFTMOUSE button until both extreme colors are visible in the *interp* window.)

4. Press MIDDLEMOUSE; the system interpolates between the colors you have chosen and places the new values into the color map.

Pixel Access Program

mag is an interactive program that magnifies or enlarges a pixel region on the screen. To use *mag*:

1. Attach to the *mag* window.
2. Place the cursor on some image on the screen, then press and release the LEFTMOUSE button. A magnified image of the area around the cursor fills the *mag* window. Pressing the MIDDLEMOUSE button toggles the outlining of the magnified pixels.

3D Object Manipulation Demonstration Programs

This section describes the fast 3D demonstration programs. They illustrate 3D transformations, polygon fill, depth-cuing, hidden surface removal, and overlapping windows. These demonstration programs are double-buffered for smooth motion.

Arch, *Flow*, *Heme*, *Jet, and Shuttle*

arch is a simulated architectural model in which the eye perceives a 3D environment. One building (the white one) is always displayed. Several other buildings can be displayed in wireframe or as solid objects.

Once you select a window for input, you interact with the program through both the keyboard and the mouse. Each mouse button or combination of mouse buttons causes a different motion of the eye within the environment. The `F` key toggles the other buildings; `S` toggles the polygon fill on the buildings. The `M` key starts a constant motion that continues even after the window is no longer selected for input (detached).

flow is an array of nearly 20,000 data points in a frame. The data was calculated at NASA-Ames Research Center on the Illiac-IV computer. *flow* presents some solutions to the Navier-Stokes equations for flow in a turbulent fluid. LEFTMOUSE controls viewer altitude and azimuth. MIDDLEMOUSE controls distance and twist (rotation on the z-axis). RIGHTMOUSE starts a sequence of small rotations. The sequence continues even if the window is detached.

heme is a depth-cued molecular model made up of 1022 points and 549 vectors. It is controlled by a popup menu. To see the menu, hold down MIDDLEMOUSE. To select or de-select an entry, position the cursor on the entry and press and release LEFTMOUSE. Two menus appear. The left column of menu entries independently turns off and on each of the five parts of the molecule. The right column of menu entries controls the transformations of the molecule. The menu entries in the right column are graphical icons.

A LEFTMOUSE press changes the color of a menu entry and starts that motion. Another LEFTMOUSE press restores the initial state. Simultaneously specified opposite motions cancel each other out, leaving no motion in the specified direction. The `ENTER` or `M` key causes current motion to continue even after the program's window is detached.

The bungalow icon (shaped like a house) resets the program back to the center of the window (especially helpful if the molecule has been dragged out of sight). The stop sign icon exits the program and cancels the window.

jet is a wireframe model of a fighter plane. The image includes 660 vectors and can be displayed with constant intensity or depth-cuing. Pressing one of the three mouse buttons enables rotation about one of the three axes. The amount of rotation is determined by the left-to-right motion of the mouse. Other combinations of mouse buttons allow *x-y* translation controlled by the mouse and scaling at fixed positive and negative rates. The `ENTER` button enables and disables depth-cueing. Press `M` to start or stop a continuous motion that continues after the window is detached.

The commands to manipulate the *shuttle* and control depth-cuing are the same as those for *jet*. Any alphanumeric key starts or stops the doors flopping open and closed. The motion does not continue after the program is detached.

Shaded Object and Curved Surface Demonstration Programs

The IRIS uses special microcode and hardware to draw Gouraud-shaded, z-buffered images rapidly. The Geometry Engines provide the capability to manipulate splines in real-time. The surface design and rendering demonstration package shows off these features. It allows you to modify the basis points that determine a curved, wireframe surface in one window and pass them to another window where a smooth-shaded, z-buffered rendering of the surface is made. The package consists of two programs: a surface editor and a rendering program.

Surface Editor

Start the surface editor window by using any of these commands: *surfcar*, *surfegg*, *surfabstract* or *surfjet*. Any of these commands runs the same program with different sets of basis points. Before you can interact with the surface editor, select the window for input.

The surface editor has two parts. The upper part of the display is the wireframe surface and the control points that define it. The bottom of the window is a menu of seven screen iconic buttons.

The simplest use of the surface editor allows one to edit the displayed surface by manipulating its basis points. To do this:

1. Move the cursor (now shaped like the mouse) clear of the window.

2. Press the mouse buttons one at a time and observe the glyphs. LEFTMOUSE gives a stylized arrow in the *x* direction, MIDDLEMOUSE gives one in the *y* direction, and RIGHTMOUSE gives one in the *z* direction.

3. Move the cursor into the top half of the display. Pick a basis point with the mouse, select a direction to move it, push the corresponding mouse button, and move the mouse from left to right to move the point. As the basis point is moved, the portion of the surface directly affected by the point is modified in real-time.

4. Now look at the menu which is lower in the display. The leftmost two screen buttons are for viewing the surface. The rightmost five screen buttons are for editing the surface. The leftmost screen button, the one showing three mutually perpendicular axes, is for rotating the surface 90 degrees around a particular axis. Move the cursor over the leftmost screen button and press and release LEFTMOUSE. The surface rotates 90 degrees around the *x-axis*.

5. Press MIDDLEMOUSE while the cursor is over the same button. The surface rotates around the *y-axis*. Pressing the RIGHTMOUSE causes the same motion around the *z-axis*. Do not try to start a second rotation until the first one is complete and the hourglass cursor has been replaced by the arrow.

6. Now select the second screen button, the bloodshot eye. Press and release LEFTMOUSE. Mouse motions now rotate the surface. Press RIGHTMOUSE to exit this mode and return the surface to its original spot.

7. Pick the eye icon button again and press and release RIGHTMOUSE. Now the surface, along with the array of index planes, rotates as guided by the mouse. RIGHTMOUSE gets you out of the rotate mode and leaves the display at its new position.

8. Use the third and fourth buttons (counting from left to right) to further subdivide the basis points that comprise the surface into twice as many basis points as before. More basis points allow for more precise editing of the surface, but also increase the complexity of the model and therefore slow it down.

9. Use the two rightmost screen buttons to make more modifications to the surface. The fifth screen button is unused. Each of the two rightmost buttons, the cubes with slices down the middle, selects a plane of symmetry (in a different axis) across the surface. When a plane is chosen (by pressing a mouse button as the cursor lies over one of the iconic buttons), half of the wireframe surface is made invisible. After a plane of symmetry has been chosen, any changes to a basis point on the visible side is also made to a corresponding point on the invisible side.

   To make both sides visible again, move the cursor over the button corresponding to the present axis of symmetry and press a mouse button.

Rendering Program

After following the procedures for manipulating the basis points of an object as outlined above, you can use the following procedure to render the surface as a Gouraud-shaded, z-buffered image in another window.

1. Move the cursor over the second menu button, the bloodshot eye. Press and release RIGHTMOUSE. Previously you have used this mode to rotate the surface and the index planes with the mouse.

2. Press and release LEFTMOUSE. The cursor changes shape to prompt you to anchor this newly created window to the screen.

   A wireframe outlining the basis points appears. The database of basis points is sent to the new window. The surface is subdivided into small quadrilaterals and an illumination model is calculated. The surface is displayed as an array of dots.

3. Detach from the surface editor window and select the rendering window for input. Rotate the object by pressing the mouse button corresponding to the axis of rotation and moving the mouse from left to right. Because the operation is double-buffered, the motion seems smooth.

4. To switch the display to splines, type `s`. To return to the point array, type `p`. To turn depth-cuing off or on, type `d`. Use `z` to Gouraud-shade the object.

   When you switch from the point or spline array to the shaded picture, the system switches itself from double to single buffering to accommodate z-buffering. Z-buffering is used for general-purpose hidden-surface removal.

5. Type `c` to select a new color ramp to smooth shade all objects on the screen. Another way to change the surface color is to bring up the *cedit*, *showmap* and *interp* graphics windows and use them to alter the color ramp that shades the surface.

The rendering program can also be executed directly without going through the surface editor stage to modify the database. To do this, type: *zshadecar*, *zshadeegg*, *zshadeabstract* or *zshadejet*.

NOTES: Hardware z-buffering is not available on machines that do not have 32 bits of image memory per pixel. Do not try to use the rendering window without a 32 bitplane system.

If a double-buffered program not set up to work with the rendering demonstration programs is currently open, the rendering program does not shade the object. It prints a warning message in the console window. Kill the outstanding double-buffer windows and try again to shade the object.

9.2 Gifts

The contents of the directory */usr/people/gifts* varies with the current software distribution. Gifts are provided as interesting and possibly useful examples, but they are not supported and are often not documented.

At the time of this printing, */usr/people/gifts* contained programming examples, window manager demonstration programs, and sample image files. Some of the programming examples are discussed in the Programming Examples section of the *IRIS User's Guide*. Some of the window manager tools are documented in the *UNIX Programmer's Manual, Volume IA*. Manual pages for window manager commands are labeled 1W.

To determine what is currently available in */usr/people/gifts*:

1. Change your working directory to the directory containing the programs:

```
cd /usr/people/gifts
```

2. Display a listing of the directories contained in this directory:

```
ls
```

Appendix A: IRIS Specifications

The standard configuration for IRIS series 3000 systems includes these components:

- Electronics cabinet with one disk drive
- Fifteen-inch or nineteen-inch tilt-and-swivel 60-Hz non-interlaced monitor
- Keyboard with right-hand and left-hand connectors for the mouse
- Optical mouse with grid pad
- Ethernet transceiver

The size specifications for each of these components is listed in Table A-1.

IRIS series 3000 systems can also be ordered with these hardware options:

- Dial and button box
- Nineteen-inch 33-Hz interlaced monitor
- Digitizer tablet
- Mitsubishi G500 Color Printer/Plotter
- Versatec ECP42 Color Plotter
- Tektronix 4692 Color Plotter
- Seiko CH-5300 Color Printer
- Half-inch tape drive

- Quarter-inch cartridge tape drive
- Stereo optic viewer cable
- Floppy disk (standard on IRIS terminal)

Early IRIS series 2000 models with the Turbo upgrade may not have the same standard peripherals described here. The 60-Hz monitor may be different from the one described in this guide. The mouse may be mechanical rather than optical. The early IRIS 2400 Turbo system required a separate junction box. For a description of these features, see the *IRIS Workstation Guide Series 2000, Version 1.0*.

See Chapter 7 for more information on optional peripherals.

Component	Height	Width	Depth
Cabinet	30.0"	18.0"	27.0"
15" Monitor	14.2"	14.4"	15.9"
19" Monitor	21.9"	19.9"	21.3"
Keyboard	1.5"	19.0"	8.5"
Mouse	1.0"	2.0"	3.0"
Transceiver	2.0"	7.0"	4.0"

Table A-1: IRIS Component Specifications

A.1 Cables

The IRIS is supplied with the following cables for connecting components:

- Four bundled, color-coded, coaxial video cables connect the video output of the cabinet to the monitor.
- Two 10-foot, 3-wire, grounded AC power cables provide power for the IRIS monitor and cabinet.

A.2 Monitor

The IRIS can drive several types of monitors. See Chapter 7, "Video Options", for a discussion of all possible video options. The monitors available from Silicon Graphics are high-resolution 60-Hz non-interlaced or 33-Hz interlaced. A single system can support two video options. The video display configuration for each system is specified when the system is ordered.

A.2.1 60-Hz Non-interlaced Monitors

The 60-Hz non-interlaced monitors have a monitor control panel on the lower right front. On the back of the monitors are several ports for receiving video signals and a power socket. These monitors are available in two sizes, nineteen-inch and fifteen-inch. (See Figures A-1 and A-2.)

A.2.2 33-Hz Interlaced Monitor

A 33-Hz interlaced monitor is an optional monitor for the IRIS system. This monitor has a monitor control panel on the right front. On the back of the monitor are several ports for receiving video signals and a power socket. (See Figure A-3).

A.2.3 Monitor Control Panel

The monitor control panel has adjustment controls and a `Power` switch.

Nineteen-inch Monitor (60-Hz Non-interlaced)

The monitor control panel on this monitor is shown in Figure 2-7.

- The button labeled `Brightness` adjusts the DC levels of the red, green, and blue signals equally. Turning this button clockwise increases the monitor's brightness. You can push the button into the monitor once the brightness is adjusted.

- The button labeled `Contrast` adjusts the AC gain levels of the red, green, and blue signals. Turning this dial clockwise increases the monitor's contrast. Once the contrast is adjusted, you can push the button in so that it is flush with the surface of the monitor.
- The button labeled `Degauss` demagnetizes the monitor screen.
- The rocker switch labeled `Power` controls power to the monitor.
- The power light indicates that the monitor power is on.

Fifteen-inch Monitor (60-Hz Non-interlaced)

The monitor control panel on this monitor is shown in Figure 2-8.

- The recessed dial labeled `Brightness` adjusts the DC levels of the red, green, and blue signals equally. Turning this knob clockwise increases the monitor's brightness.
- The recessed dial labeled `Contrast` adjusts the AC gain levels of the red, green, and blue signals. Turning this dial clockwise increases the monitor's contrast.
- The button labeled `Degauss` demagnetizes the monitor screen.
- The button labeled `Power` controls power to the monitor.
- The power light indicates that the monitor power is on.

33-Hz Interlaced Monitor

The monitor control panel on the 33-Hz interlaced monitor is shown in Figure 2-9.

- The knob labeled `Brightness` adjusts the DC levels of the red, green, and blue signals equally. Turning this knob clockwise increases the monitor's brightness.
- The knob labeled `Contrast` adjusts the AC gain levels of the red, green, and blue signals. Turning this knob clockwise increases the monitor's contrast.

- The button labeled `Degauss` demagnetizes the monitor screen.
- The switch labeled `Power` controls power to the monitor.
- The light labeled `Health` lights when power to the monitor is switched on.

A.2.4 Monitor Back Panel

The monitor back panel has several connectors for the cables that connect the monitor to the cabinet (see Figures A-1, A-2, and A-3).

Nineteen-inch Monitor (60-Hz Interlaced)

- Either of the two BNC sockets labeled `Sync` receives the video sync signal from the cabinet.
- Either of the two BNC sockets labeled `R`, `G`, and `B` receives the red, green, or blue video signal from the cabinet.
- The fuse protects the circuit. See the information printed next to the fuse for type and rating requirements. (See Figure A-1.)
- The power selector switch selects the power voltage.
- The power input receptacle attaches to the power cable. See the information printed next to the receptacle for power requirements. (See Figure A-1.)
- The four 75 Ω terminators attached to the monitor by chains control the impedance of the video signals. When a single monitor is used, the terminators plug into the unused BNC sockets. When more than one monitor is used in a daisy-chain, the terminators are removed from the sockets on all but the last monitor.

Fifteen-inch Monitor (60-Hz Non-interlaced)

- Either of the two BNC sockets labeled `Sync` receives the video sync signal from the cabinet.
- Either of the two BNC sockets labeled `R`, `G`, and `B` receives the red, green, or blue video signal from the cabinet.
- The fuse protects the circuit. See the information printed next to the fuse for type and rating requirements. (See Figure A-2.)
- The power selector switch selects the power voltage. See the information printed on the switch for power requirements. (See Figure A-2.)
- The power receptacle attaches to the power cable.
- The four impedance switches control the impedance of the video signals. When the switch is `IN`, it is in the `75 Ω` position. When the switch is `OUT`, it is in the `High` position.

33-Hz Interlaced Monitor

- Either of the two BNC sockets labeled `Ext Sync` receives the video sync signal from the cabinet.
- The two BNC sockets labeled `V D` are not used.
- Either of the two BNC sockets labeled `R`, `G`, and `B` receives the red, green, or blue video signal from the cabinet.
- The circuit protector is either a 5-amp fuse (100-125 VAC) or a 2-amp fuse. (200-240 VAC).
- The 120/240 volt power receptacle connects to the power cable.
- The 25-pin plug connects the control cable from the cabinet to the monitor.
- The four impedance switches control the impedance of the video signals.

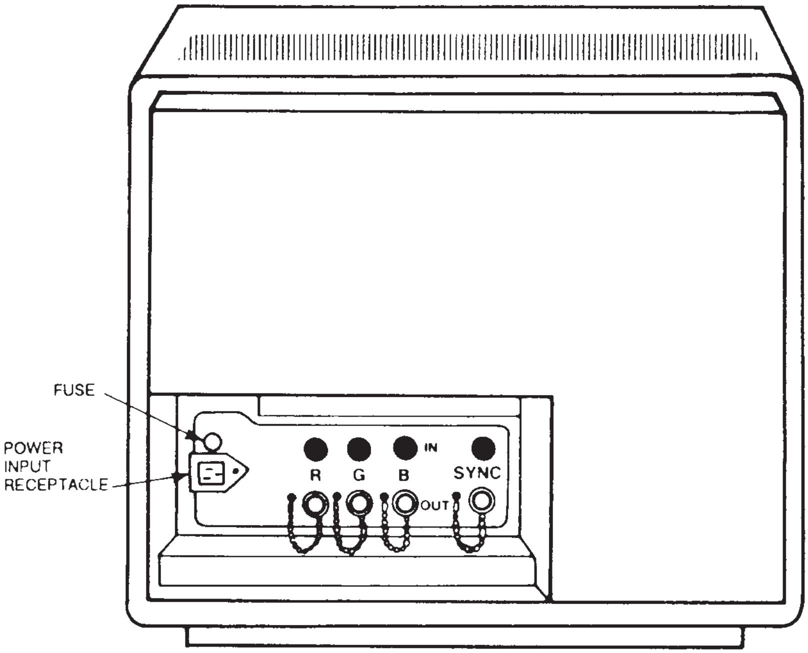


Figure A-1: Monitor Back Panel for Nineteen-inch 60-Hz Monitor

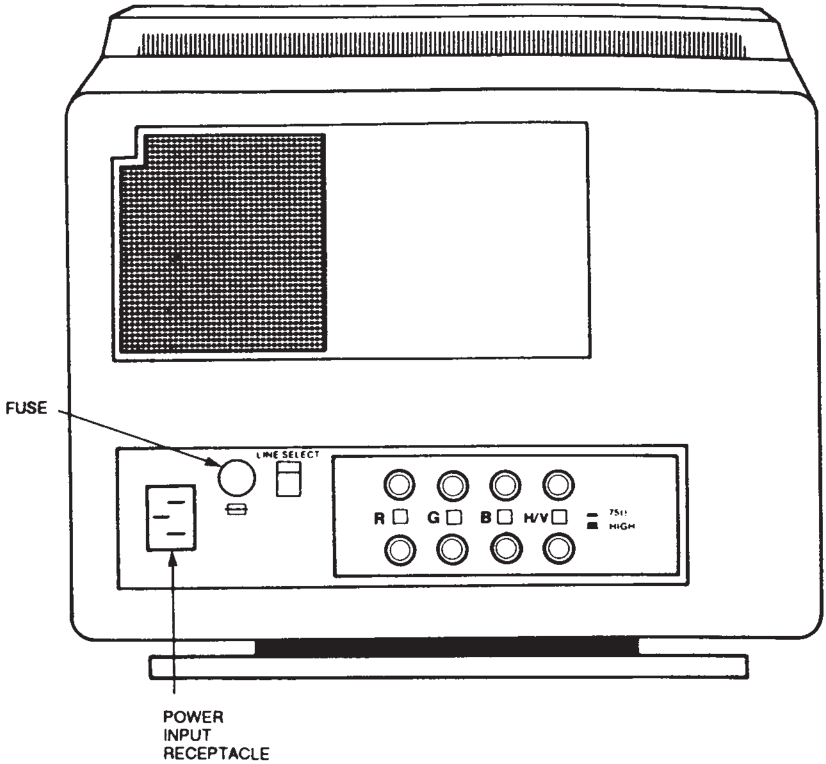


Figure A-2: Monitor Back Panel for Fifteen-inch 60-Hz Monitor

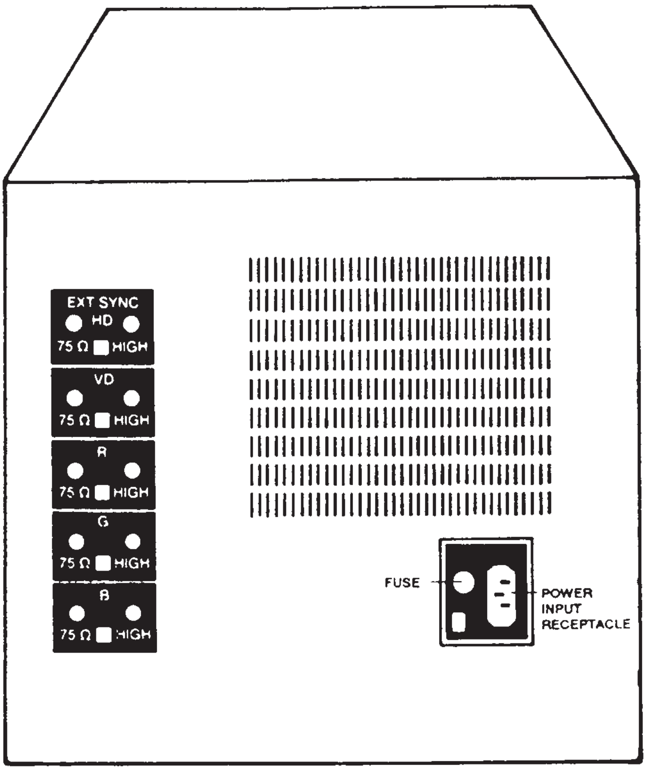


Figure A-3: Monitor Back Panel for 33-Hz Interlaced Monitor

A.3 IRIS Electronics Cabinet

There are three panels on the back of the IRIS cabinet: a standard I/O panel, an auxiliary I/O panel, and a power panel (see Figure A-4). On the front of the IRIS is a cartridge tape drive (optional) and a `Power` switch that controls power for the IRIS system.

A.3.1 Power Switch

The `Power` switch for the IRIS is located on the front upper left corner of the cabinet. This switch controls power for the cabinet. It does not control the power for any auxiliary equipment connected to the cabinet through the convenience outlet located on the cabinet power panel.

A.3.2 Optional Tape Drives

Both a 60 Mb quarter-inch cartridge tape drive and a half-inch tape drive are offered as optional components on the IRIS.

Quarter-inch Cartridge Tape Drive

The cartridge tape drive is located on the front upper left of the cabinet. See Chapter 7, "Optional Peripherals", for information on using the tape drive.

Half-inch Tape Drive

The half-inch tape drive is housed in an enclosure that is separate from the electronics cabinet.

A.3.3 Standard I/O Panel

The standard I/O panel is located on the upper right rear of the cabinet (see Figure A-4). This panel has ports for connecting the monitor, a network drop cable, a floppy disk drive, and various RS-232 devices.

- The RS-232 connector labeled `Port 1` attaches to the keyboard.
- The RS-232 connectors labeled `Port 2`, `Port 3`, and `Port 4` are available for RS-232 or RS-423 serial lines.
- The 15-pin D socket labeled `Ethernet` connects the IRIS to an Ethernet drop cable.
- The BNC socket labeled `Sync` provides the video sync signal for the monitor.
- The three BNC sockets labeled `Red`, `Green`, and `Blue` provide the red, green, and blue video signals for the monitor.
- The `Reset` button resets the processor, which in turn resets the rest of the system. After the `Reset` button has been pressed, the IRIS either reboots automatically or waits for boot instructions.

  CAUTION: Do not press the `Reset` button while the IRIS is running UNIX. For information on rebooting the system, see Chapter 3. If the IRIS is not running UNIX and is under control of the PROM monitor, then the `Reset` button or the `Power` switch may be used. See the discussion on Crash Recovery in Chapter 4.
- The LED labeled `Halt` lights when the processor is in a halt state.
- The alphanumeric diagnostic LED labeled `Status` indicates system status and displays startup diagnostics.
- The nine-element DIP switch labeled `Configuration` controls the IRIS' startup diagnostics, and the boot environment.

A.3.4 Cabinet Power Panel

The cabinet power panel has a power inlet receptacle (see Figure A-4).

- The male 3-pin input power receptacle labeled `Power` accepts power for the IRIS system.
- For operation in the US, a 20 amp, 250V, 3AB normal-blow ceramic tube fuse protects the circuit. For European operation, a 6.3 amp, 250V, type T slow-blow fuse protects the circuit.

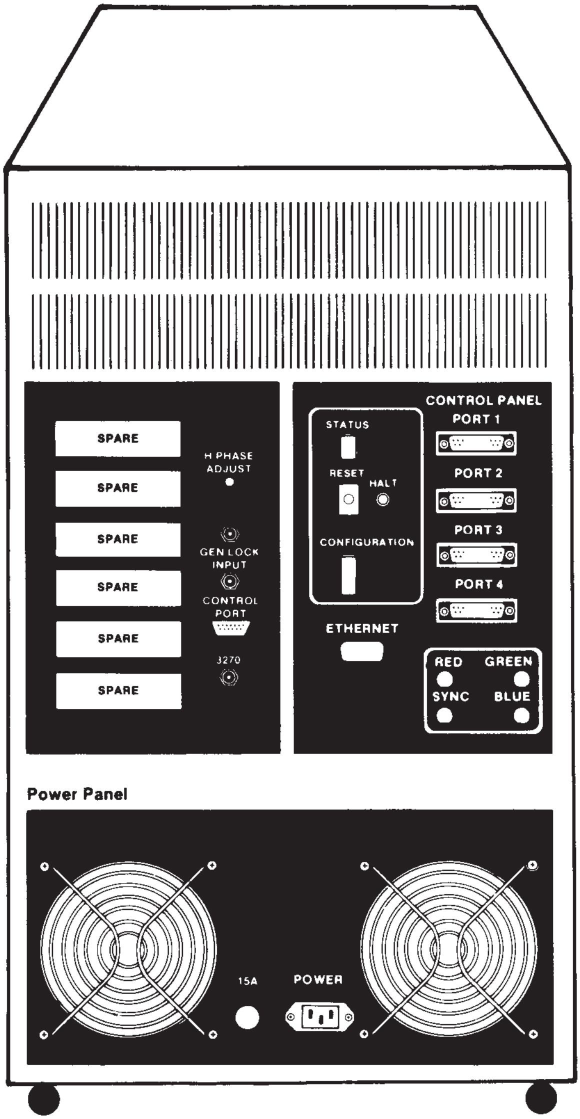


Figure A-4: IRIS Cabinet Back Panel

A.4 Site Selection

Table A-2 contains a list of guidelines for site selection for your IRIS. Although site selection is the customer's responsibility, Silicon Graphics' representatives will provide consulting services upon request.

Temperature	50 — 86°F (operating) 32 — 122°F (non-operating)
Relative humidity	20 — 80%
Minimum clearance	3" all sides
Monitor desk space Nineteen-inch Hz	 30" width 30" depth
Fifteen-inch	 22" width 22" depth
Cabinet space	24" width 33" depth 33" height
System Power Requirement	single phase line, neutral, ground
115 VAC Configuration Operating Range	 93-132 VAC 47-63 Hz
220 VAC Configuration Operating Range	 186-264 VAC 47-63 Hz
System Power Consumption	Cabinet 1250 VA/1000 W Nineteen-inch 60-Hz Monitor 225 VA/150 W Fifteen-inch Monitor 185 VA/100 W Nineteen-inch 33-Hz Monitor 170 VA/116 W
Heat dissipation	Cabinet 3412 BTU/hr Nineteen-inch 60-Hz Monitor 512 BTU/hr Fifteen-inch Monitor 341 BTU/hr Nineteen-inch 33-Hz Monitor 395 BTU/hr
Card Slots	20

Table A-2: IRIS Environmental Specifications

Appendix B: System Messages

When the UNIX kernel on an IRIS workstation reaches an unrecoverable error condition, it displays an error message preceded by the word `panic:`.

If the error message includes one of the abbreviations listed in Table B-1, the condition is probably caused by a hardware problem. Table B-1 lists the devices associated with each abbreviation. The abbreviated device names are usually followed by a digit indicating a physical unit or letters referencing the function that lead to the problem.

Abbreviation	Device
si	IRIS 3030 disk drive
mt	tape drive (any type)
qic	quarter-inch cartridge tape
md	IRIS 3010/3020 hard disk drive
mf	floppy disk drive
dsd	tape, small disk, and floppy disk controller
nx	Ethernet

Table B-1: Device Abbreviations

If the panic message does not indicate one of the devices listed in Table B-1, there may be a hardware problem with another part of the system (processor board, memory board, or backplane, for example) or a software problem in the kernel.

Anytime you get a panic message, try to run *sync* and reboot (see Section 3.1 or 3.2). Run *fsck* to check the file system. If the message reappears, call the Geometry Hotline (see Chapter 1).

NOTE: Do not interrupt (e.g., by pressing `Ctrl-c`) processes that are accessing the no-rewind tape device. Attempts to do this always result in a kernel panic.

B.1 Hardware Error Messages

The following sections describe hardware error messages.

B.1.1 General Hardware Error Messages

- **IO err in swap**

  While swapping a user process, a hard error occurred on the swap disk.

- **parity error**

  A parity error occurred in the system memory. Another message preceding this error message indicates where in physical memory the error occurred. Unix cannot diagnose the memory failure.

- **iinit**

  The system was unable to read the root file system. Either the disk drive or the root file system is damaged.

B.1.2 Disk Controller Messages

IRIS 3010/3020 Disk Controller Messages

A. The following messages result from a bad disk controller.

- **md[0-1][a-h]: couldn't start!**
- **dsd: zero status on command**
- **dsd: ccb timeout, "command" dev=0 unit=[0-1]**
- **dsd: ccb timeout during init**
- **dsd: cib timeout, "command" dev=0 unit=[0-1]**

B. This message indicates that no label is installed on the disk.

- **(***No label***)**

C. The following error probably results from a bad connection from the disk to the controller, a bad disk, or hard disk errors encountered when one reads or writes to the disk.

- **dsd0: hard error, "command" dev=0 unit=[0-1]**

IRIS 3030 Disk Controller Messages

A. This message indicates that no label is installed on the disk.

- **(***No label***)**

B. The following messages indicate that a hard error has occurred and recovery may not be possible.

- **"Read" or "Write" on si[0-4], slice [a-h]**
- **siintr: sitab.b_active == 0**
- **siintr hard error NN: "error" block: NN cmd: command**

C. The following messages occur in non-interrupt mode and could mean a faulty controller, a bad set of disk cables, or incorrect jumpering of the controller.

- **sicmd: timeout wait for status**
- **sicmd: timeout waiting for cmd "command" to complete**
- **sicmd: status error: "error"**
- **sicmd: error stat "status" err "error" iostat "iostatus"**

D. The following error indicates a recoverable problem with the disk. This message may appear when there is a bad track on the disk. If this message appears repeatedly, contact Customer Service through the Geometry Hotline (see Chapter 1).

- **soft error "error" on block NN**

B.1.3 Tape Drive Messages

A. The following are general messages for the tape drive.

- **qic0: no cartridge in drive**

  Cartridge isn't inserted properly.

- **qic0: unit not ready**
- **qic0: write protected**

  Dial on the cartridge needs to be twisted to enable writing on the tape.

B. The following error message indicates that the tape drive has no power or that the control cable for the tape drive is disconnected or reversed.

- **qic0: can't get status during init**

C. The following error messages indicate that there is a problem with the tape drive that inhibits its operation. The operation using the tape drive must be restarted.

- **qic0: couldn't start**
- **qic0: hard error during "Read or Write" status= XX**
- **qic0: "command" failed**

8.1.4 Floppy Controller Messages

A. The following are general messages.

- **mf0: write protected**
- **mf0: disk formatted**
- **Reading on mf0**
- **Writing on mf0**

B. The following message indicates a problem with the floppy disk controller.

- **mf0a: hard error, cmd='command',error='error'**

B.2 System Software Messages

The following error messages are system software messages.

A. The following error messages indicate a problem with the Ethernet hardware or software.

- **nxpresent cleared**
- **xns_ttstart**

B. The following message occurs while the system is booting. This message indicates that the system attempted to configure a disk drive with no entry in the *bdevsw[]* array (the array of block devices). The system was incorrectly configured.

- **getmajor**

C. The following messages indicate that there is a problem with the buffer cache/inode tables.

- **devtab**
- **no fs**
- **no imt**

D. The following message indicates that the *init* process was killed. If the process was killed through a user program or the shell *kill* command, then this message may not reflect a problem. If the *init* process was killed during the boot procedure, then this message indicates a problem with the root file system.

- **init died!**

E. The following message occurs when the system attempts to put a time-driven event on its queue, and there is no room remaining. If this occurs frequently, then the system has been incorrectly configured.

- **timeout table overflow.**

F. The following messages indicate problems with the kernel.

- **trap**
- **kernel address error**
- **kernel bus error**
- **ksyscall**

G. The following messages are related to graphics consistency checks.

- **duplicate gr_getshmem**

- **gr_unlockmem**

H. The following messages indicate problems occurring while trying to swap some portion of a user process to disk.

- **IO err in push**

- **hard IO err in swap**

I. The following message occurs during the autoconfiguration process if no disk was found that supports a swap area.

- **no swap device.**

J. The following message indicates that all available swap space has been used.

- **swkill.**

K. Error messages not listed above are related to internal kernel consistency checks.

Appendix C: A Printing Primer for the IRIS

This appendix provides a general introduction to printing documents on the IRIS Workstation. Such documents may include (depending on the printing option you have selected) color or grey-shaded screen images, laser typeset text, and simple ASCII file dumps. The information presented in this appendix is applicable to all printers currently supported by Silicon Graphics, Inc. for use with the IRIS. Descriptions of operations and features that are specific to a given printer model may be found in the documentation accompanying the peripheral option package supporting that printer.

This appendix addresses two major topics:

- Printer management using the LP system
- Preparation of output files

There is also a section on adding a dumb printer.

C.1 Using the LP Spooling Utilities

The Unix Line Printer (LP) Spooling Utilities is a set of eleven commands that allow you to "spool" a file that you want to print. *Spooling* is the name given to the technique of temporarily storing data until it is ready to be processed (in this case, by your printer). For LP spooling, a file (or group of files) to be printed is stored in a queue until a printer becomes available. When the printer is ready, the next file in the queue is printed.

LP spooling allows you to make use of your workstation without having to wait until your file is printed. LP spooling also allows one printer, or several printers, to be shared among many users. The flow of printing throughout your system is regulated by the LP Spooling Utilities.

The LP Spooling Utilities allow:

- Customizing your system so that it will spool to a pool of printers. These printers need not be the same type.
- Grouping printers together into logical classes to maximize throughput.
- Queueing print requests, thus allowing a print request (job) to be processed by the next available printer.
- Cancelling print requests, so that an unnecessary job will not be printed.
- Starting and stopping LP from processing print requests.
- Changing printer configurations.
- Reporting the status of the LP scheduler.
- Restarting any printing that was not completed when the system was powered down.

The eleven line printer spooling commands are divided into two categories: *user* commands are for general use of the LP system; *administrative* commands are for system configuration and maintenance.

C.1.1 Definitions and Conventions

The following terms used in this appendix represent important concepts in the LP system:

device Depending on context, either an apparatus for obtaining printer output or to one of several special "device" files located in */dev*. UNIX uses these files to access peripherals such as printers, terminals, and tape drives.

printer A logical name that represents a physical device, i.e., the actual printer.

class The name given to an ordered list of one or more printers. A printer may be assigned to more than one class, but need not be assigned to any class.

destination The place an LP request is sent to await printing. The destintation may be a specific printer or a class of printers. An output request sent to a specific printer will be printed only by that printer; a request sent to a class of printers will be printed by the first available printer in its class.

Figure C-1 illustrates the conventions used within the screen examples in this appendix.

```
This type style is used to show system responses
generated by the IRIS.
```

`Bold type style is used to show keyboard input.`

[KEY] Boxes are used to show non-printing keystrokes.

Figure C-1: Screen Example Conventions

C.1.2 User Commands

This section summarizes the five basic LP commands. A detailed description of each command follows the summary.

User Command Summary

lp	Routes jobs to a destination and places them in a queue. The destination may be either a single printer or a class of printers.
cancel	Cancels output requests.
disable	Prevents a printer from processing jobs in the queue.
enable	Allows a printer to process jobs in the queue.
lpstat	Reports the status of all aspects of the LP Spooling system.

lp: Make an Output Request

The *lp* command routes a job request to a destination where it is placed in a queue to await printing. The destination may be a single printer or a class of printers. If no destination is specified, the request is routed to the default destination. For information on how to set the default printer destination, see section C.1.4, "General Management".

The form of the *lp* command is:

`lp` [*options*] *file(s)*

Every time an *lp* request is made, a "request ID" is assigned to the job, and a record of the request is sent to you. This request has the following form:

destination – seqnum

destination is the printer or class of printers to which the job has been routed. *seqnum* is an arbitrary sequence number assigned to the job by the LP system.

lp has three options which are particularly useful: **–n**, **–d**, and **–c**.

Use **–n** to print more than one copy of a document:

`lp -n`*number*

number is the number of copies to print.

Use **–d** to specify a printer or class of printers other than the default printer (assuming your system is connected to more than one printer):

`lp -d`*destination*

Finally, use **–c** to ensure that no changes will be made to your file once you have issued a print request.

These command options may be combined in any order. For a complete list of *lp* options, see the entry for *lp*(1) in the *UNIX Programmer's Manual, Volume IA*.

```
% lp myfile  RETURN
request id is myprinter-12 (1 file)

% lp < myfile  RETURN
request id is myprinter-13 (standard input)

% cat myfile |lp  RETURN
request id is myprinter-14 (standard input)

% lp -n3 -dfoo -c myfile  RETURN
request id is foo-15 (1 file)

%
```

Figure C-2: lp command Examples

There are several different ways to request a printout with the *lp* command. The first three examples in figure C-2 perform identical functions, sending a simple print request to the default printer. The fourth example prints three copies on printer "foo", and creates a copy of the file for the printer to process, thus ensuring that no changes are made to the file after the print request.

cancel: Stop a Print Request

The *cancel* command removes a job from the queue. It can be invoked either before or after a job has started printing, but only one job can be cancelled at a time.

Any user may cancel any other user's job. If you cancel another user's request, mail is sent to that user. Once a job is cancelled, it can be requested again only with the *lp* command.

The *cancel* command takes two forms:

`cancel` *printer-name*

`cancel` *request-ID*

Cancelling using the printer name cancels the job currently being printed. Using the request ID cancels the specified job whether or not it is currently being printed. Figure C-3 illustrates examples of the *cancel* command.

```
% cancel myprinter  RETURN
request "myprinter-16" cancelled

% cancel myprinter-17  RETURN
request "myprinter-17" cancelled

%
```

Figure C-3: cancel Command Examples

Issuing a *cancel* command will not work when the job is being printed on a remote machine. To cancel requests on a remote machine, see the section "Printing to a Remote Device" later in this appendix.

disable: Stop Printer from Processing Requests

The *disable* command prevents the printer from processing jobs in the queue. Possible reaons for disabling the printer include malfunctioning hardware, paper jams, running out of paper, or end-of-day shutdowns. If a printer is busy at the time it is disabled, the request it was printing is reprinted in its entirety when the printer is re-established.

Job requests can be sent to a printer that has been disabled. The jobs are put on the queue but are not printed until the printer is enabled.

The *disable* command takes the form:

```
disable [-c] [-r"reason"] printers
```

The **–c** option cancels the request currently being printed as well as disabling the printer. This is useful if the output is causing the printer to behave abnormally.

The **–r** option lets you tell other users why the printer was disabled. *reason* is a character string, and is enclosed in double quotes ("). This string is reported to anyone trying to use the disabled printer.

Figure C-4 illustrates an example of the *disable* command.

enable: Allow Printer to Process Requests

The *enable* command permits a printer that has been disabled to begin processing jobs from the queue. The *enable* command takes the form:

```
enable printers
```

Figure C-4 illustrates an example of the *enable* command.

```
% disable -r"paper jam" myprinter RETURN
printer "myprinter" now disabled

% enable myprinter RETURN
printer "myprinter" now enabled


%
```

Figure C-4: disable and enable Command Examples

lpstat: Report LP Status

The *lpstat* command gives you a report on the status of various aspects of the LP system. The *lpstat* command takes the form:

`lpstat [`*options*`]`

The most useful option is **–t**, which gives a complete report on the status of the LP system. For a complete list of options, see the entry for *lpstat*(1) in the *UNIX Programmer's Manual, Volume IA*. If you are printing on a remote printer (i.e., one that you are accessing via a network, and not directly), see the section on remote printing in this appendix for information on how to get remote status reports.

Figure C-5 illustrates use of the *lpstat* command:

```
% lpstat -t  RETURN
scheduler is running
system default destination: myprinter
members of class foo:
        myprinter
device for myprinter: /usr/spool/lp/etc/myprinter-log
myprinter accepting requests since Jul 31 21:40
foo accepting requests since Jul 30 12:23
printer myprinter now printing foo-18.
enabled since Aug 5 15:34
foo-18     mylogin    3156   Aug 7 17:11 on myprinter
%
```

Figure C-5: lpstat Command Example

C.1.3 Adminstrative Commands

This section summarizes the commands that are used to administer the LP system. To execute the administrative commands, you must be logged in as either "root" (i.e., as the super-user) or "lp". Use of administrative commands by inexperienced users is not recommended.

Administrative Command Summary

lpsched Starts the LP scheduler.

lpshut Stops the LP scheduler.

reject Prevents jobs from being queued at a particular destination.

accept Permits job requests to be queued at a particular destination.

lpmove Moves printer requests from one destination to another.

lpadmin Configures the LP system.

lpsched: Start the LP Scheduler

The *lpsched* command starts the LP scheduler. As long as the scheduler is running, jobs requested by *lp* will be printed. *lpsched* is executed automatically each time the IRIS is booted.

Every time *lpsched* is executed, it creates a file called *SCHEDLOCK* in */usr/spool/lp*. As long as this file exists, the system will not allow another scheduler to run. When the scheduler is stopped under normal conditions, *SCHEDLOCK* is removed. If the scheduler should stop abnormally, remove *SCHEDLOCK* before attempting to use the *lpsched* command. This procedure may also be necessary to restart the scheduler after the system has been taken down abnormally.

The *lpsched* command takes the form:

```
/usr/lib/lpsched
```

There is no response from the system to acknowledge the *lpsched* command; to verify that the scheduler is running, use *lpstat*.

lpshut: Stop the LP Scheduler

Occasionally, it is necessary to stop the LP scheduler, as when reconfiguring the system using the *lpadmin* command. *lpshut* stops the LP scheduler and ends all printing activity. All requests that were being printed when the command was issued are reprinted in their entirety when the scheduler is restarted.

The *lpshut* command takes the form:

```
/usr/lib/lpshut
```

reject: Prevent Print Requests

Sometimes it is necessary to stop *lp* from routing requests to a destination queue. For example, if a printer has been removed for repairs, or has received too many requests, you may wish to prevent new jobs from being queued at that destination. The *reject* command performs this function.

Requests in the queue at the time *reject* is issued will be printed as long as the printer is enabled. The *accept* command allows requests to be received again.

The *reject* command takes the form:

`/usr/lib/reject [-r"`*reason*`"]` *destination(s)*

The **–r** option lets you tell other users why print requests are being rejected. *reason* is a character string, and is enclosed in double quotes ("). This string is reported to anyone trying to use *lp* to send requests to the specified destination.

accept: Allow Print Requests

The *accept* command allows job requests to be placed in a queue at the named printer(s) or class(es) of printers. The *accept* command takes the form:

`/usr/lib/accept` *destination(s)*

Figure C-6 illustrates examples of the reject and accept commands.

```
% su RETURN

# /usr/lib/accept myprinter RETURN
destination "myprinter" now accepting requests

# /usr/lib/reject -r"printer broken" myprinter RETURN
destination "myprinter" is no longer accepting requests

#
```

Figure C-6: accept and reject Command Examples

lpmove: Move a Request to Another Printer

Occasionally, you may find it necessary to move output requests from one destination to another. For example, if you have a printer removed for repairs, you may want to move all jobs pending on the queue to a destinatinon with a working printer. This is done with the *lpmove* command. *lpmove* may also be used to move specific requests from one destination to another, but only once the scheduler has been halted with the *lpshut* command. *lpmove* will automatically reject job requests re-routed to a destination without a printer.

The *lpmove* command takes two forms:

`/usr/lib/lpmove` *dest1 dest2*

`/usr/lib/lpmove` *request(s) destination*

dest1, *dest2*, and *destination* are printers or classes of printers. *request* is a specific request ID.

In the first form of the command, all requests are moved from *dest1* to *dest2*. After the move, the printer or printers at *dest1* will not accept requests until an *accept* command has been issued. All re-routed requests are renamed *dest2*-nnn.

In the second form, which may be issued only after the scheduler has been halted, the re-routed requests are renamed *destination*-nnn. When the scheduler is re-started, the original destinations will still accept new requests.

Figure C-7 illustrates examples of the *lpmove* vommand.

```
%  su RETURN

#  /usr/lib/lpmove myprinter yourprinter RETURN

#  /usr/lib/lpmove foo-19 foo-20 yourprinter RETURN
total of 2 requests moved to yourprinter

#
```

Figure C-7: lpmove Command Examples

lpadmin: Configure Printers

The *lpadmin* command has two primary uses: adding new printers to the system, and changing printer classes and destinations. Since Silicon Graphics supplies routines to automatically add the printers supported for use with the IRIS, the options for adding printers are useful only in the case of dumb printers. These options are covered in section C.1.5, "Adding a Dumb Printer".

Unlike most UNIX commands, *lpadmin* requires an option. The *lpadmin* command takes three forms:

`lpadmin -d[`*destination*`]`

`lpadmin -x`*destination*

`lpadmin -p`*printer*

The **–d** option is used to set the system default destination. The *destination* chosen must already exist when the command is issued. Complete instructions on how to define the default destination are in section C.1.4, "General Management".

The **–x** option removes the specified *destination* from the LP system. This form of the *lpadmin* command will NOT work while the scheduler is running.

No destination (printer or class) may be removed if it has pending requests. All requests must either be removed with the *cancel* command or moved to other destinations with *lpmove* before the destination can be removed.

Removing the last remaining member of a class causes the class to be deleted. If the destination removed is the system default, the system will no longer have a default destination. Removal of a class, however, does not imply the removal of printers assigned to that class.

Silicon Graphics, Inc. provides a shell script, *rmprinter*, which automatically removes a printer and its associated *log* and option files. *rmprinter* is described in C.1.4, "General Management".

The **–p** form of the lpadmin command has many options, most of which are covered in section C.1.5, "Adding a Dumb Printer". There are two options that allow re-assignment of printers to different classes. With these options, the lpadmin command takes the form:

`lpadmin -p`*printer* `[-c`*class*`] [-r`*class*`]`

The **–c** option assigns a printer to the specified *class*; the **–r** option removes a printer from the specified *class*.

The **–p** options will NOT work while the scheduler is running. For a complete list of options, see the entry for *lpadmin*(1M) in the UNIX *Programmer's Manual, Volume IA*.

Figure C-8 illustrates examples of the lpadmin command.

```
%  su  RETURN
#  /usr/lib/lpadmin -xmyprinter  RETURN
#  /usr/lib/lpadmin -dmyprinter -rfoo -cboo  RETURN
#
```

Figure C-8: lpadmin Command Examples

C.1.4 General Management

This section includes information on the following topics:

- Setting the default printer destination
- Rotating log files
- Printing to a remote device
- Removing a printer

Setting the Default Printer Destination

The *lp* command determines the destination of a request by checking for a **–d** option on the command line. If no **–d** is present, it checks to see if the environment variable LPDEST is set. If LPDEST is not set, then the request is routed to the default destination.

The system default destination can be a printer or a printer class. It is set by using the *lpadmin* command with the **–d** option. The system default MUST be set by the user; it is not done automatically by the printer installation software that Silicon Graphics supplies. A destination must already exist on the LP system before it can be assigned as the default destination.

Setting the environment variable LPDEST allows a user to have a default destination rather than the system default.

Figure C-9 illustrates examples of setting the system default with *lpadmin* and setting the user default with LPDEST.

```
% su RETURN
# /usr/lib/lpadmin -dmyprinter RETURN
#

% env LPDEST=yourprinter RETURN
%
```

Figure C-9: Setting Default Printers with lpadmin and LPDEST

Rotating *log* Files

The purpose of a *log* file is to keep a record of all printing activity on a given printer. Each printer has a separate *log* file, located in */usr/spool/lp/transcript/log* if the printer is an Apple LaserWriter, and in */usr/spool/lp/etc/log* otherwise. The name of each printer's *log* file takes the form:

printer-name–log

Each file contains a running list of processed jobs, each of which includes the following:

- the *logname* of the user who made the request
- the request ID
- the name of the printer that processed the request
- the date and time that the printing started.

Any *lpsched* error messages that occur are also recorded.

If there is a large number of LP requests for a given printer, that printer's *log* file will soon get very large. You can manually remove the contents of these files from time to time, or you can set up the IRIS to do it for you automatically at regular intervals.

Included in */usr/spool/lp/etc/util* is a shell script *log.rotate* which will automatically rotate (clean out) your printers' log files. To set up the script for your printer(s), you must edit *log.rotate* in the following manner:

1. Become the super-user (with the *su* command).

2. Remove the comment marker (#) from the following line:

```
#printers="PRINTER1 PRINTER2"
```

3. In place of `PRINTER1` and `PRINTER2`, put the names of any parallel-interface (i.e., color) printers, and any remote printers. Any number of printers may be included. If you have no color or remote printers, use the null string ("") in place of printer names.

4. Remove the comment marker (#) from the following line:

```
#LocalPS="PRINTER1"
```

5. In place of `PRINTER1`, put the names of any hard-wired (i.e., connected to the serial port) LaserWriters. If you do not have a hardwired LaserWriter, use the null string ("") in place of printer names.

Printing to a Remote Device

Remote printing on the IRIS allows users to send print jobs over the network with the same commands they use to send jobs to a printer connected directly to their IRIS. This is accomplished by giving the remote printer a local name that the LP scheduler is "fooled" into thinking is a local printer. After LP puts a job request for the remote printer into the queue the request is sent across the net to the remote machine, whose LP system takes over the request. As a result of this, one cannot accurately determine the status of a remote print request by using the *lpstat* command on the local machine.

This section covers the two major aspects of remote printing:

- Installing a remote printer on the LP system
- Getting status on remote print requests
- Cancelling remote print requests

Installing a Remote Printer

A shell script for installing remote printers on the LP system is included in */usr/spool/lp/etc/util* and is called *mknetpr*. *mknetpr* is executed as a command line and takes the form:

`./mknetpr` *printer netaddr netprinter*

printer is the local name you want for the remote printer, *netaddr* is the name of the machine the remote printer is on, and *netprinter* is the name of the printer on that machine.

You must be the super-user to execute *mknetpr*.

If your IRIS is using TCP/IP network protocols, you will have to perform the following procedure:

On the remote machine, execute the shell script *addclient*, in */usr/spool/lp/etc/util*. *addclient* takes the form:

`./addclient` *local-machine-name*

You must be the super-user to execute *addclient*.

You will also need to edit the */etc/hosts* file on BOTH machines so that they will be aware of each other's presence on the net. Information on */etc/hosts* can be found in the *TCP/IP User's Guide*.

Getting Remote Printer Status

When a print request is sent across the net to a remote machine, the local LP system will always report that request as being printed, regardless of its actual status in the remote machine's LP system. It is therefore necessary to remotely access (using *rlogin*, *xlogin*, *xx*, etc.) the machine whose printer is processing the job to get the job's status. The remote LP scheduler changes the request ID of any job sent to it over the net to reflect the actual name of the printer, and gives it a new sequence number corresponding to its place in the remote queue. The way to determine a specific job's status is to look in the remote printer's *log* file (i.e., the *log* file on the remote machine) with the *tail* command:

`tail` *logpath*

logpath is the pathname of the remote printer's log file.

Cancelling Remote Print Requests

Once you have the remote printer status, you can use the *cancel* command on the remote machine to cancel any desired job or jobs on the printer's queue. A remote print job must be cancelled from the remote machine once it has been sent over the net.

Removing a Printer

Under some circumstances, you may want to remove a printer entirely from the LP system. A shell script named *rmprinter* is provided in */usr/spool/lp/etc/util* for this purpose. It removes all *log* and option files for a given printer. *rmprinter* takes the following form:

`./rmprinter` *printer-name*

You must be the super-user to execute *rmprinter*.

C.1.5 Adding a Dumb Printer

The purpose of this section is to provide some guidance on how to connect a serial dumb printer to your IRIS.

Silicon Graphics does not currently support the use of dumb printers with the IRIS Workstation. This does not mean that they cannot be attached, it simply means that you will have to configure and interface such a printer yourself. Only users familiar with printer interfacing should attempt to connect a dumb printer.

LP Administrative Procedures

The following set of procedures provides a general outline for adding a dumb printer to the LP system:

1. Become the super-user (with the *su* command).

2. To avoid unwanted output from non-LP processes and to ensure that LP can write to the device, enter the following commands:

   ```
   chown lp /dev/ttydport
   chmod 600 /dev/ttydport
   ```

   port is the port number to which the printer is connected.

3. To prevent the IRIS from sending a login prompt to the printer, edit the file */etc/inittab* in the following manner:

   Find the line corresponding to the serial port your printer is connected to. It should look something like this:

   ```
   dport::respawn:/etc/getty ttydport dx_9600
   ```

   Add an '**x**' after the first colon, changing the line to the following:

   ```
   dport:x:respawn:/etc/getty ttydport dx_9600
   ```

   port is a number one less than the number of the port to which the printer is connected (i.e.,' 1' for port 2, etc.).

4. Turn off the scheduler with the *lpshut* command.

5. Introduce the printer to the LP system with the *lpadmin* command. You need to specify a *printer* name, a *device* file, and an interface program or *filter*.

   The *printer* name must conform to the following rules:

   - It must be no longer than ten characters
   - It must consist solely of alphanumeric characters and underscores
   - It must be unique within the LP system

   device is the pathname of the UNIX System device file associated with the printer, */dev/ttyd0*, */dev/ttyd1*, */dev/ttyd2*, or */dev/ttyd3*, depending on which serial port the printer is connected to.

   The *filter* is chosen with either of the following options: **–m** or **–i**.

   Use **–m** if you want to use one of the "model" filters supplied with the LP system in */usr/spool/lp/model*.

   `lpadmin -p`*printer* `-v`*device* `-m`*model*

   See the following subsection, "Printer Filters" for more on the model filters.

   Use **–i** if you want to use a filter that you have written yourself:

   `lpadmin -p`*printer* `-v`*device* `-i`*filter*

   filter is your own filter.

   You may also want to use the **–h** option, which tells LP that the printer is hardwired to the IRIS, **–h** does not take any arguments, and may appear anywhere after the **–p** option on the command line.

6. Start the LP scheduler with the *lpsched* command.

7. Allow the printer to accept requests to its queue with the *accept* command.

8. Enable the printer with the *enable* command.

Printer Filters

Printers that are used with the LP system must have a printer interface program, or *filter*. Every print request made with the *lp* command is routed through an appropriate filter before it is printed, as illustrated in figure C-10.

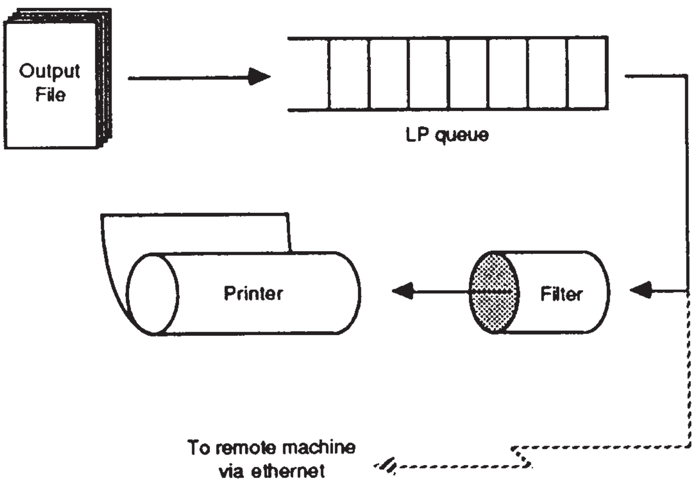


Figure C-10: Path of a Print Request Through the LP System

The *lpadmin* command is used to assign a filter to a printer.

A number of "model" interface programs in the form of shell scripts are provided in */usr/spool/lp/model*, including a generic dumb printer interface, *dumb*. Edit this script to meet the particular needs of your printer. The following information should help you to create your own printer filter.

When LP routes an output request to a printer, the filter for the printer is invoked by LP in the directory */usr/spool/lp* as follows:

`interface/`*printer id user title copies options file*

The filter takes the following arguments:

printer	printer name
id	request ID returned by *lp*
user	logname of the user who made the request
title	optional title specified by the user
copies	number of copies requested by the user
options	list of class- and printer-dependent options specified by the user
file	full pathname of a file to be printed

When the filter is invoked, its standard input comes from */dev/null*, and both the standard output and standard error output are directed to the printing device.

Filters format their output by using command line arguments. Your filter must have the proper terminal characteristics (such as baud rate) included in its command lines. You may do this by adding lines containing the *stty* command in the following form:

`stty` *options* `<&1`

This command takes the standard input for stty from the device. For more information on the *stty* command, see the entry for *stty*(1) in the *UNIX Programmer's Manual, Volume IA*. Since different printers have different numbers of columns, make sure that the header and trailer for your filter correspond to your printer's specifications.

When printing is complete, it is the responsibility of your filter to exit with a code that shows the status of the print job.

Exit codes are interpreted by *lpsched* as follows:

0 The print job has completed successfully.

1-127 A problem was encountered in printing the request. *lpsched* notifies the sender by mail that there was a printing error. Subsequent jobs are not affected.

> 127 These codes are reserved for internal use by *lpsched*.

When problems occur that are likely to affect subsequent print jobs you should have your filter disable the printer so that requests are not lost. When a busy printer is disabled, the filter exits with code **15**.

C.2 Preparing Output Files

Two kinds of files are used to generate printed output on the IRIS: text files and image files.

C.2.1 Text Files

Text files consist of ASCII characters, and are generally created using a text editor such as *vi*, which is documented in the *UNIX Programmer's Manual, Volume IIA*. Color printers can not print ASCII text files.

Plain text can be printed directly on the Apple LaserWriter or on dumb printers with the *lp* command. (Note: the LaserWriter is NOT able to print ASCII files without the software supplied by Silicon Graphics, Inc. with the laser printer option package.)

Laser-typeset text can be produced on the Apple LaserWriter using the *Documenter's Workbench* software supplied with the laser printer option. For more information on laser-typesetting, see the *UNIX System V Documenter's Workbench Software User's Guide*, the *UNIX System V Documenter's Workbench Software Technical Discussion and Reference Manual*, and *Using Your Apple LaserWriter*, which are supplied by Silicon Graphics, Inc. with the laser printer option.

C.2.2 Image Files

Image files consist of representations of screen pixels. There are three routines found in */usr/people/gifts/mextools/imgtools* that provide a straightforward means of creating image files from the screen: *capture*, *snap*, and *savemap*. Each of these routines is provided as source code; to create a executable version of the routine, run *make* on the source file.

The *capture* routine creates an image file of the entire screen. *capture* takes the following form:

`capture` *ifile*

ifile is the image file to which the screen data is written. *capture* must be used from within the window manager.

The *snap* routine creates an image file of a selected portion of the screen. *snap* takes the following form:

`snap` *ifile* [*xsize* [*ysize*]]

ifile is the image file to which the screen data is written. *xsize* and *ysize* are optional dimensions that specify the size of the region in pixels that is to be "grabbed". If no size is specified, the user is free to select the size of the screen area by manipulating a window "frame" on the screen. If a size is specified, the frame is of set dimension, but can be moved anywhere on the screen. *snap* and other useful graphical routines are described in the *IRIS User's Guide, Volume II*, "Appendix E: Window Manager Programs". *snap* must be used from within the window manager.

Before printing an image file created with *capture* or *snap*, you should also create a *color map* file to send to the printer. This is done with the routine *savemap*. *savemap* takes the following form:

`savemap` *mapfile*

mapfile is the file to which the color map data is written.

To print your image files, issue a command line of the following form:

`lp` *ifiles mapfile*

The files will be printed one to a page; the images will be scaled to fit. (Note: it is necessary to include a color map file when printing screen images with the Apple LaserWriter as well as with color printers, to ensure proper grey-scaling of the black-and-white image.)

Information on more complex editing of image files can be found in the *IRIS User's Guide, Volume II*," Appendix H: Using the Image Library".

All printers supported by Silicon Graphics, Inc. for use with the IRIS can print image files.

Appendix D: IRIS Serial Ports and Cabling

The serial ports on the IRIS behave differently depending on the version of software running and model type of the IRIS. This appendix outlines the serial support on series 2000 Turbo series 3000 workstations and provides instructions for connecting serial devices to the ports for systems running GL2-W3.5 software.

D.1 Defining the Serial Interface

The IRIS workstation provides an RS-232 serial interface that drives between +4.4 and –4.4 Vdc. All RS-232 cables that you connect to the IRIS should be shielded. The IRIS has no problem driving and receiving signals on a 50-foot cable, and it typically drives and receives signals on an RS-232 cable up to 200 feet long.

There are two types of serial interface equipment available. These are the *Data Terminal Equipment* (DTE) and the *Data Communications Equipment* (DCE). The primary difference between DTE and DCE is the designation of several pins on the connector. For example, DTEs transmit on pin 2 and receive on pin 3. DCEs transmit on pin 3 and receive on pin 2. You can connect a DTE interface directly to a DCE interface.

To connect either a DCE to a DCE, or a DTE to a DTE, use a null modem cable. A null modem cable has the wires to pins 2 and 3 swapped in one connector, and may have other swapped wires as well. A signal on pin 2 at one end appears on pin 3 at the other end, and vise-versa.

The IRIS workstation serial ports are all configured as DTE. Most terminals are also configured as DTE. Therefore, to connect a terminal to the workstation, use a cable that has pins 2 and 3 swapped in one connector. To connect a modem to the workstation, use a RS-232 cable that connects each pin of the IRIS serial port to the corresponding pin of the modem. No signals need to be swapped. Connect other peripheral devices according to the configuration data provided with the device.

Silicon Graphics, Inc., provides three kinds of special files, which determine which driver is used on each port. The special files *ttyd1*, *ttyd2*, and *ttyd3* are used for devices such as terminals; the files *ttym1*, *ttym2*, and *ttym3* are used for modems, and *ttyf2* and *ttyf3* are used for flow control to devices that understand hardware flow control.

GL2-W3.5 Software		
Type	*Port*	*Device*
2	ttyd1	2,3,7,20
3	ttyd2	2,3,7,20
4	ttyd3	2,3,7,20
2	ttym1	2,3,4*,7,8,20
3	ttym2	2,3,4*,7,8,20
4	ttym3	2,3,4*,7,8,20
2	-	-
3	ttyf2	2,3,4*,5,7,8,20
4	ttyf3	2,3,4*,5,7,8,20

Table D-1: IRIS series 3000 and 2000 Turbo Device Types and Pins

*Pin 20 is positive when the device is open.
Pin 4 is the same as pin 20 for *ttyd[1,2,3]* and *ttym[1,2,3]*.
Pin 4 means "o.k. to send" for *ttyf[1,2,3]*,

ttyd[1,2,3]=device, ttym[1,2,3]=modem, ttyf[1,2,3]=flow control

If you have an IRIS 2400 Turbo, IRIS 2500 Turbo, or any series 3000 workstation, you have a Motorola 68020 (IP2) processor. Table D-1 shows the pins and devices the IRIS supports on each port of the I/O panel if the IRIS is running version GL2-W3.5 software.

Use `Port 1` only for the system console on all IRIS workstation models.

Table D-2 shows the signals the IRIS supports on each port.

IP2		
Port	*Abbreviation*	*Signal*
2	TD RD DCD DTR	Transmit data Receive data Data carrier detect Data terminal ready
3	TD RD DCD DTR RTS CTS	Transmit data Receive data Data carrier detect Data terminal ready Request to send Clear to send
4	TD RD DCD DTR RTS CTS	Transmit data Receive data Data carrier detect Data terminal ready Request to send Clear to send

Table D-2: Current Processor RS-232 Support

IP2 is a Motorola 68020 processor

D.2 Cabling the Serial Ports

This chapter describes the cables typically used to connect the IRIS to terminals, printers, and modems.

Serial devices are connected through `Port 2`, `Port 3`, or `Port 4` on the cabinet I/O panel.

The serial ports on the IRIS are designed to connect directly to *Data Communications Equipment* (DCE) devices such as modems, via a modem cable. Each wire on this cable connects the same pin of the IRIS to the modem, i.e., it has a pin-to-pin correspondence. Table D-3 shows the pin definitions for the IRIS and for the modem cable. These pin definitions are supported for devices *ttym1*, *ttym2*, or *ttym3* on an IRIS running GL2-W3.5 software.

Modem Cable		
IRIS	*Modem*	*Signals*
1	-	Chassis ground
2	2	Transmit data
3	3	Receive data
4	4	Request to send
5	5	Clear to send
8	8	Data carrier detect
20	20	Data terminal ready
7	7	Signal ground

Table D-3: Pin Definitions for Modem Cable

Connecting the IRIS to *Data Terminal Equipment* (DTE) devices, such as terminals and printers, requires a different cable arrangement, a *null modem cable*. A null modem cable has the wires for pin 2 and pin 3 swapped in one connector, and may have other wires swapped as well. Table D-4 lists the pin definitions for an example of a null modem cable. The pin numbers that are shown separated by commas (,) should be connected together and also to the other pin or pins listed in the same row.

Sample Null Modem Cable		
IRIS	*Terminal*	*Signals*
1	-	Chassis ground
2	3	Transmit data
3	2	Receive data
4	8	Request to send, Clear to send
8	4,5*	Data carrier detect
6	20	Data set ready
20	6.22*	Data terminal ready
7	7	Signal ground

*These connections may be necessary on the terminal side.

Table D-4: Pin Definitions for a Null Modem Cable

These pin definitions work with any serial device that complies with the RS-232 specification.

Most printers work with simpler cabling. Refer to your printer manual for pin specifications, because many printers are different Table D-5 shows the signals the IRIS supports on the port

IRIS		
pin	*signal*	*abbrev.*
1	Chassis ground	PG
2	Transmit data	TD
3	Receive data	RD
4	"caught up" for *ttyf[1,2,3]*, "I'm here" for *ttym[1,2,3]*	RTS
5	"o.k. to send"	CTS
7	Signal ground	SG
8	The other end is alive (if it's not connected, assumes the other end is alive)	DCD
20	"I'm here"	DTR

Table D-5: Pin Signals for the IRIS and Printer

Table D-6 shows the pins typically used by printers.

Printer	
pin	*signal*
4, 6, 8, or 20	"I'm alive"
5, 6, or 20	"I'm caught up"

Table D-6: Pin Signals Used Typically by Printers

Most terminals do not require the various handshaking lines such as *clear to send* or *data set ready*, and will work with a *three-wire null modem cable*. The signals for pins 2 and 3 must be swapped, and pin 7 of the IRIS connects to pin 7 of the terminal. Table D-7 lists the pin definitions for a three-wire null modem cable.

Sample Three-wire Null Modem Cable		
IRIS	*Terminal*	*Signals*
2	3	Transmit data
3	2	Receive data
7	7	Signal ground

Table D-7: Sample of a Three-wire Null Modem Cable for Terminals

Table D-8 summarizes the types of cables to use with different peripherals.

Cable Usage	
Peripheral	*Cable*
Modem	Modem cable
Other computer	Null modem
Printer	Simplified null modem
Terminal	Three-wire null modem

Table D-8: Summary of Cable Types