TEK SYSTEM MANUAL

Part No. 070-5329-01
Product Group 65

6130 System

SYSTEM ADMINISTRATION

First Printing DEC 1984
Revised APR 1986

Tektronix®
COMMITTED TO EXCELLENCE

WARNING

This equipment generates, uses, and can radiate radio frequency energy and if not installed and used in accordance with the instruction manual, may cause interference to radio communications. It has been tested and found to comply with the limits for Class A computing devices pursuant to Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such interference when operated in a commercial environment. Operation of this equipment in a residential area is likely to cause interference in which case the users at their own expense will be required to take whatever measures may be required to correct the interference.

This software and documentation is based in part on the Fourth Berkeley Software Distribution under license from the Regents of the University of California. We acknowledge the following individuals and institutions for their role in its development: The Regents of the University of Caifornia, the Section on Medical Information Science at the San Francisco Campus of the University of California, the Electrical Engineering and Computer Sciences Department at the Berkeley Campus of the University of California, and to the Department of Mathematics and Computer Science of the Vrije Universiteit, Amsterdam, the Netherlands.

WARNING

This equipment generates, uses, and can radiate radio frequency energy and if not installed and used in accordance with the instruction manual, may cause interference to radio communications. It has been tested and found to comply with the limits for Class A computing devices pursuant to Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such interference when operated in a commercial environment. Operation of this equipment in a residential area is likely to cause interference in which case the users at their own expense will be required to take whatever measures may be required to correct the interference.

This software and documentation is based in part on the Fourth Berkeley Software Distribution under license from the Regents of the University of California. We acknowledge the following individuals and institutions for their role in its development: The Regents of the University of California, the Section on Medical Information Science at the San Francisco Campus of the University of California, the Electrical Engineering and Computer Sciences Department at the Berkeley Campus of the University of California, and to the Department of Mathematics and Computer Science of the Vrije Universiteit, Amsterdam, the Netherlands.

Contents

SECTION 7 SYSTEM HEALTH

SECTION 8 SYSTEM HALTS

SECTION 9 SYSTEM RECONFIGURATION

Figures

Examples

Tables

Safety Summary

Symbols on Equipment

 ATTENTION — refer to manual.

Terms

In This Manual

CAUTION statements identify conditions or practices that can result in damage to equipment or other property.

Marked on Equipment

CAUTION indicates a personal injury hazard not immediately accessible as one reads the marking, or a hazard to property including the equipment itself.

Use the Proper Power Cord

Use only the power cord and connector specified for your product.

Use only a power cord that is in good condition.

Refer cord and connector changes to qualified service personnel.

Power Source and Ground

The 6100 and 6200 Series workstations are designed with a protective ground connection in the Tektronix-supplied power cord. A protective ground connection by way of the grounding connector in the power cord is essential for safe operation. To avoid electrical shock, plug the power cord into a properly wired outlet.

This product is designed to operate from a power source that does not apply more than 250 volts rms between the supply conductors or between either supply conductor and ground.

Use Care When Accessing Back Panel

When you access the back panel (to change boards, attach connectors, check line voltage or configuration switch settings, or whatever) FOLLOW ALL DIRECTIONS CAREFULLY. Always shutdown and unplug the system at the point and in the manner that the instructions describe.

Do Not Remove Covers or Panels

To avoid personal injury, do not remove the workstation's covers or panels, unless instructed to do so by the manual. Do not operate the workstation without the cover and panels properly installed.

1

Introduction

THE 6130 INTELLIGENT GRAPHICS WORKSTATION

The 6130 is a workstation for a professional engineering, design, or office environment. It was created as a general-purpose machine with many options, so it can be easily adapted to meet the needs of different groups of users.

The 6130 workstation has a UNIX-based operating system, called UTek, which contains a set of integrated programming and documentation tools: text processing, electronic mail and two-way communication, a variety of editors, source code control, debuggers, compilers, and support of the C, FORTRAN 77, Pascal, and BASIC programming languages.

System Box

The 6130 *system box* is the cabinet and hardware components that make up the workstation. The system box is small enough to fit on or under a standard desk. The standard cabinet contains a 40-megabyte Winchester hard disk drive and a 5.25-inch diskette (flexible disk) drive. You can order a 6130 with an 80- or 120-megabyte Winchester disk, instead of the standard 40-megabyte disk.

The standard workstation contains one megabyte of random access memory, two asynchronous RS-232-C ports, a Local Area Network interface (IEEE Standard 802.3 compatible), and a GPIB interface. Any port-compatible terminal or peripheral can be connected to the workstation, but you are responsible for making sure the configuration of the terminal or peripheral matches the communications protocols of the workstation (discussed in this manual).

Enhancements

Tektronix offers several options and enhancement products for the 6130 workstation. You can add expansion memory and various interfaces, including SCSI and additional GPIB and RS-232-C interfaces.

The peripherals compatible with the 6130 workstation that are available through Tektronix include additional diskette drives, streaming cartridge tape drives, electrostatic printers or plotters, and dot-matrix printers.

As your needs grow, you can connect your 6130 workstation to other workstations or to a large computer using a Local Area Network (LAN).

This manual also discusses how to modify your 6130 workstation system to deal with the enhancements and peripherals currently available from Tektronix, or to use the LAN.

Diskette Distribution

When you get your 6130 workstation, the UTek operating system is already present on the Winchester disk. You should also have received a set of nine diskettes. These are:

- The *standalone utilities* diskette, containing utilities that can run without UTek. These include **saformat**, which formats diskettes and the Winchester disk, and **sacopy**, which copies data between devices when UTek is not available.
- The four *miniroot* diskettes, which, when copied to the Winchester disk, contains the minimum system required for the system to boot. However, you cannot boot the system with only the miniroot; you also need a kernel.
- The *miniroot system* diskette, which contains a copy of the UTek kernel. Use this diskette to boot the system if you are using the miniroot system, or if the kernel on the Winchester disk is not usable.
- The four *system configuration* diskettes, which let you configure the kernel by using the *sysconf* interface. You should configure the kernel when you first install the workstation. For more information, see Section 9 of this manual.

Keep these diskettes in a safe place. Guard them not only from harm, but also from theft, since no 6130 can be kept secure from someone who has copies of these diskettes.

The use of these diskettes is discussed in Section 8 of this manual (except for the system configuration diskettes, which are discussed in Sections 5 and 9).

ABOUT THIS BOOK

This is the *System Administration* guide for your 6130 workstation. The information in this book is designed to help you:

- Set up the workstation for the first time.
- Perform common system tasks.
- Monitor and maintain the workstation operating system.
- Administer the connection to the Local Area Network, if your workstation is connected to one.
- Deal with system halts.
- Reconfigure and tune the system as you add or remove enhancements, or as your system needs change.

Section 1 — Introduction This section introduces the 6130 workstation and explains how to use this manual.

Section 2 — First Time Start-up This section covers the procedure for bringing up the workstation for the first time, from verifying correct installation to logging in for the first time.

Section 3 — LAN Administration This section deals with connecting the workstation to a Local Area Network (LAN) and performing system administration tasks pertaining to the LAN.

Section 4 — Sysadmin Interface Procedures This section deals with the system administration operations you can perform with the menu-driven *sysadmin* interface. This interface lets you perform some of the more complicated procedures without having to know all the details about UTek. These operations include:

- System configuration maintenance.
- Performing system backups and restores.
- Adding applications and optional software.
- Adding and deleting users.
- Adding and deleting groups.

Section 5 — Other Procedures This section discusses system administration procedures that are not part of the sysadmin interface. These procedures include starting the system up and shutting it down, adding devices, and setting the time and date.

Section 6 — Concepts for System Procedures This section covers concepts for some of the procedures discussed in Sections 4 and 5. Read this section for background information about the tasks you perform as system administrator.

Section 7 — System Health This section covers the tasks you have to perform in order to keep the system operating smoothly. These tasks are mostly maintenance and monitoring activities. They include:

- Preventive maintenance tasks, such as maintaining the file system and making sure there's enough disk space for efficient operation.
- System messages reported to the console.
- Logging the tasks you perform.
- Security.
- Dealing with common nonfatal problems (troubleshooting).

Section 8 — System Halts This section covers possible causes of system halts, and recovery procedures for those causes. Both hardware and software causes for system halts are discussed.

Section 9 — System Reconfiguration This section tells you how to use the *sysconf* interface to reconfigure the UTek kernel.

Appendixes

There are also seven appendixes at the back of this manual.

A. *Fsck (file system check) Messages.* This appendix lists and explains the messages you can receive while you are running the **fsck** program to check the file system.

B. *Hardware Considerations.* This appendix contains information about the workstation hardware that you may need as system administrator, such as how to remove and replace enhancement boards.

C. *Reinstalling 64WP02 UTek.* This appendix contains the procedure for restoring the system if you did not take backups and need to totally rebuild the system. Take frequent backups and you'll never need this appendix.

D. *List of Terminal Acronyms.* This appendix contains a list of the most common terminal acronyms from the */etc/termcap* file. When configuring the workstation to be compatible with your terminal, find the UTek abbreviation for your terminal from this list.

E. *Disk Space Maintenance.* This appendix describes how to free up disk space when/if the hard disk gets close to full.

F. *UTek Recovery Procedures.* This appendix describes the step by step procedures for recovering from a system crash.

Intended Audience

This book is designed for the new administrator. You do not have to be familiar with system administration tasks. However, most of this book assumes a certain level of computer experience and a working knowledge of UTek.

No matter what your background, use Section 2 of this manual to bring up the system for the first time. Then, based on these guidelines, determine what your next step in learning UTek should be.

If you are not familiar with a UNIX-like operating system, go through the online sessions in the *6130 Learning Guide* and read *Introducing the UNIX System* by Henry McGilton and Rachel Morgan (especially Chapter 14, the UNIX System Management Guide). Also read the *6130 System User's Guide* for information on daily use of the system. Do this before you perform the system administration tasks discussed in this book.

If you are familiar with a UNIX-like system, read Chapter 14 of *Introducing the UNIX System* by Henry McGilton and Rachel Morgan before performing the system administration tasks in this book.

If you have been a system administrator on a non-UNIX system, use this book as a reference for how system administration should be performed on UTek. If you need to learn UTek, go through the online sessions in the *6130 Learning Guide* and read *Introducing the UNIX System* by Henry McGilton and Rachel Morgan and the *6130 System User's Guide* for information on daily use of the system.

If you have been a system administrator on a UNIX-like system, this manual is probably most useful to you as a reference to the sysadmin interface described in Section 4, and as a troubleshooting reference for 6130-specific problems.

Conventions

All references in this manual to indicators, connectors, switches, and so on assume that the workstation is sitting horizontally on a flat surface. When the workstation is in this position, the Start/Stop switch is in the lower right corner of the workstation front panel, and the printing near the connectors on the workstation back panel is oriented properly. If you have the workstation standing sideways in its optional floor stand, remember to consider this different position.

This manual assumes that you have a working knowledge of UTek and of basic workstation operation. If you do not know how to enter commands or use the diskette drive, refer to the *6130 Learning Guide* or the *6130 System User's Guide*. Remember to terminate each command line this manual tells you to enter by pressing <RETURN>.

This manual often refers to files using the full pathname for the file the first time, and a shorter version of the name subsequently. For example, *crontab* is the same file first introduced as */usr/lib/crontab*.

Notation

The notation conventions used in this manual are:

<RETURN>	Labeled keys are shown in all capital letters, surrounded by <angle brackets>.
<CTRL-X>	Create control characters by holding down the <CTRL> key while pressing the indicated key, such as *X*.
file	Where indicated, replace the word in *italics* with the name of your own file, directory, or path.
[*yy*]	In a command line, information inside brackets is optional; it does not have to be entered.
cd	Commands you type and responses from the system are shown in bold type when used in discussions in the text.
`MAKEDEV std`	Commands you enter are shown in `constant width` type in examples and interactive procedures.
fsck(8)	A number in parentheses after a command tells you to look in that section of the *UTek Command Reference* for more information about the command.

MORE ABOUT 6130 DOCUMENTATION

This manual is probably the first manual you will look at after the *6130 System Installation* manual. Section 2 tells you how to start the workstation and log in for the first time. However, once you have logged in, if you don't know how to use UTek, you should go to the other manuals in the 6130 documentation set to learn about the system. Figure 1-1 shows a recommended path through the documentation for system administrators of various experience levels.

These other books are available for the 6130 workstation and for the UTek operating system:

- *6130 Learning Guide.* A beginner's guide to the 6130 workstation and UTek. This book also covers how to use the online learning sessions.
- *Introducing the UNIX System*, by Henry McGilton and Rachel Morgan. A good book to read if you've never used a UNIX-based operating system before.
- *6130 System User's Guide.* Describes the 6130 system in depth — how it works, how to program for it, how to operate peripherals.
- *UTek Tools.* Contains detailed information about some aspects of the UTek operating system, including the file system and the mail system. (*UTek Tools* is made up of two volumes.)
- *UTek Command Reference.* Contains detailed information about all UTek commands. (*UTek Command Reference* is made up of two volumes.)
- *Network File System Reference.* This manual contains information about NFS and how to set up your environment.
- *6130 Quick Reference.* Summarizes details of frequently-used commands and operations.
- *Programming Language Books.* Reference books on C, Pascal, BASIC, and FORTRAN 77.

Hardware installation and service manuals are also available for the workstation.

If you need a book you do not have, contact your Tektronix Field Office.

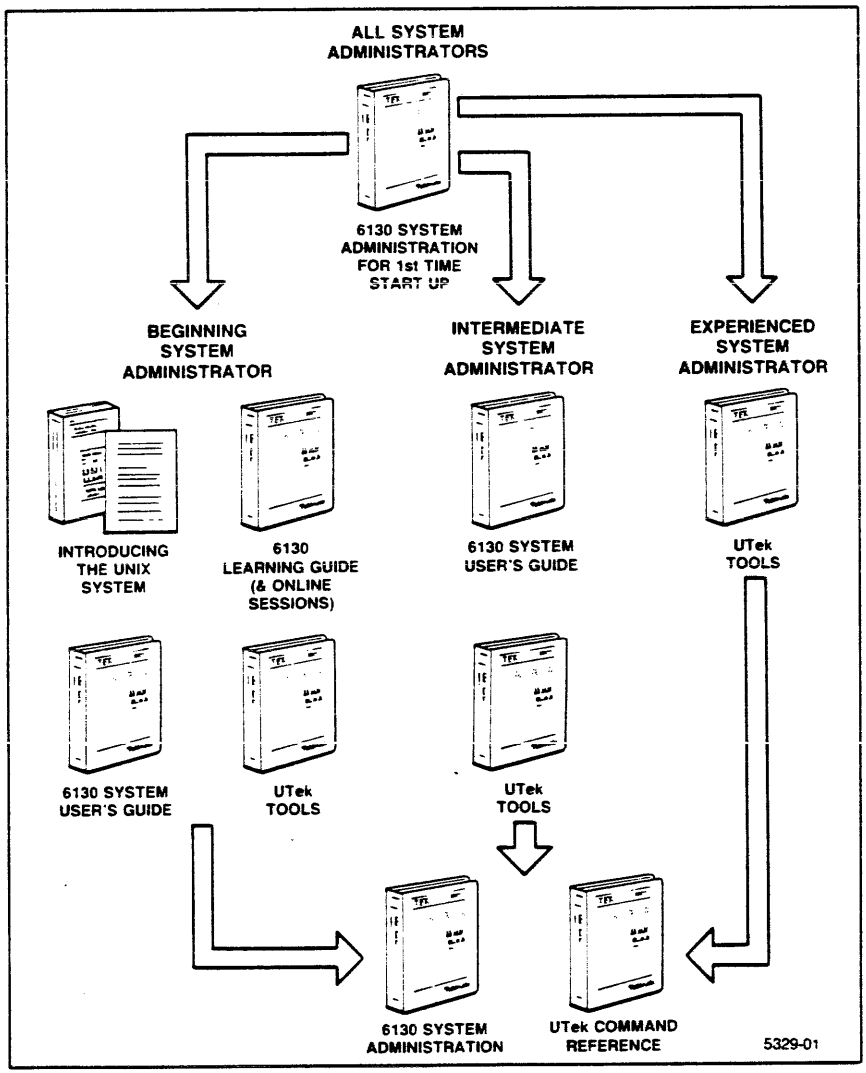


Figure 1-1. System Administrator's Documentation Path

First Time Start-Up

INTRODUCTION

This section goes through the steps you should perform the first time you start up the 6130 workstation.

The workstation and any peripherals you intend to use should be out of their boxes and set up in their proper locations. All cables should be plugged into their appropriate power receptacles. For instructions on how to unpack and connect the workstation and its peripherals, see the *6130 System Installation* manual.

GETTING READY FOR POWER-UP

Much of the following discussion refers to connectors, switches, and selectors on the back panel of the workstation. Figure 2-1 shows the back panel. To reach the back panel, unlatch and gently lift the cable management cover.

Follow this procedure before you push the start/stop switch for the first time:

1. Check that all cables are properly connected. Possible cables include:
   - Power cord (Check that the start/stop switch is off before plugging in the power cord.)
   - Cables between terminals and workstation
   - Cables between peripheral devices and workstation
   - LAN transceiver cable
2. Check that the line voltage is set to the proper level; choose between 110 volts (domestic) or 220 volts (European). The position of the yellow line voltage indicator on the left side of the workstation back panel tells you the current line voltage setting (see Figure 2-1). This indicator is not a selector. If you need to change the line voltage, contact your Tektronix Field Office.
3. Check that the eight configuration switches are properly set. Figure 2-1 shows the location of the configuration switches on the back panel of the workstation. Figure 2-2 shows a detail of the configuration switches and what each switch specifies. The switches are numbered from 1 to 8; switch 1 is on the left, and switch 8 is on the right, as you face the back of the workstation.

   NOTE

   The console terminal is used to display power-up messages when the workstation is first turned on. The factory default for a console terminal is an ANSI-compatible terminal set to 9600 baud and connected to port 1. You must have a console terminal connected to the console port or for the workstation to power up.

   At the factory, all switches have been set in the up position except switch 3, which is down. This assumes that you want autoboot, and that a 9600 baud ANSI-compatible terminal is connected to port 1 as the console device. If you don't want this configuration, see Tables 2-1 and 2-2 for other configurations you can choose.
4. Turn on the console device and all other external peripherals, such as extra terminals, printers, plotters, and so on.

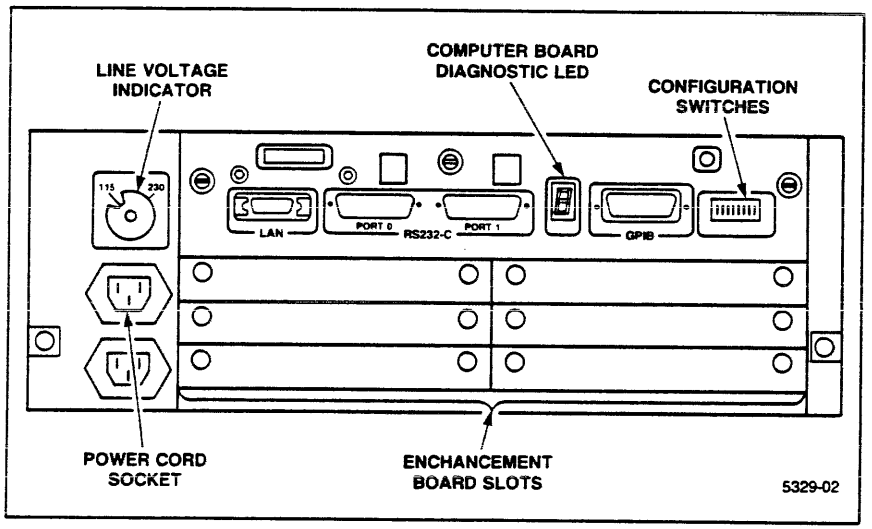


Figure 2-1. Back Panel of the 6130 Workstation

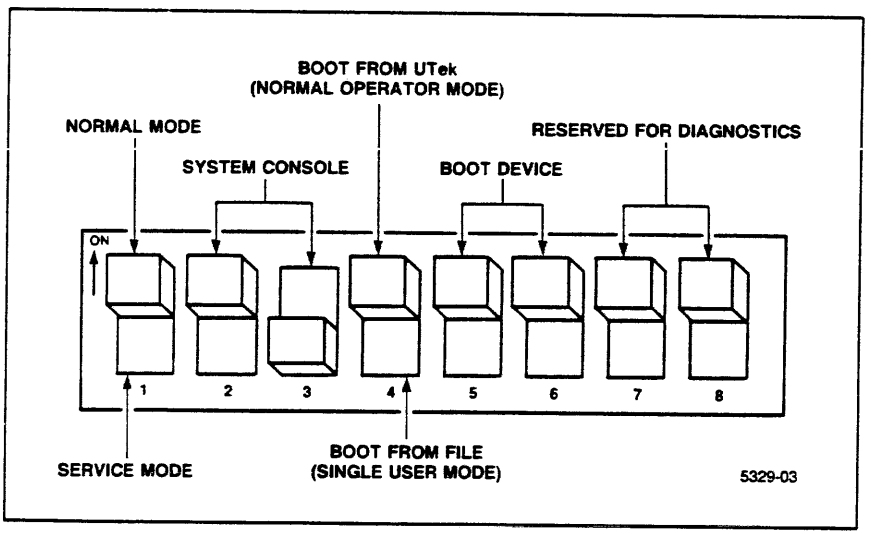


Figure 2-2. Configuration Selection Switches

Configuration Switches

Configuration switch 1 sets the workstation to either Normal (up) or Service (down) mode. The setting of switch 1 determines the meaning of the rest of the switches. Switch 1 should be set to Normal mode (up).

Switches 2 and 3 let you select the device you want as *console*. The console displays system messages. There can be only one console, and it must be selected before you turn the workstation on.

Table 2-1 shows what the settings for configuration switches 2 and 3 mean.

The setting of switch 4 determines whether the workstation boots (starts) UTek from the Winchester disk (up), or from a file that you specify (down). Switch 4 also specifies whether the workstation boots in multiuser (normal operating) mode (up), or single-user mode (down). (Bringing the system up in single-user mode using switch 4 is a subset of booting from a file.)

Switch 4 should be set to boot UTek (up).

Switches 5 and 6 specify the device that the workstation looks to for boot information, or, the *boot device*. Possible boot devices are:

- Winchester disk
- Diskette drive
- Local Area Network (LAN) port

Table 2-2 shows the settings for switches 5 and 6 to select the boot device.

Autoboot means that the workstation searches a sequence of devices from which to boot UTek. The workstation tries to boot from (in order):

1. Diskette drive
2. Winchester disk
3. Local Area Network (LAN)

Switches 5 and 6 should be set to autoboot (both up) for first time start-up.

Switches 7 and 8 are reserved for use with the Diagnostics operating system. These two switches should always be up during normal system operation.

Table 2-1
CONSOLE DEVICE SETTINGS

Console Device	Switch 2	Switch 3
undefined	up	up
9600 baud RS-232-C terminal (port 1)	up	down
1200 baud RS-232-C modem/terminal (port 0)	down	up
300 baud modem/terminal (port 0)	down	down

Table 2-2
BOOT DEVICE SETTINGS

Boot Device	Switch 5	Switch 6
Autoboot	up	up
Winchester disk	up	down
Diskette drive	down	up
LAN port	down	down

TURNING THE WORKSTATION ON FOR THE FIRST TIME

Once you have checked all the connections and switches, determined that everything is set correctly, and turned on the console device and other peripherals, it is time to power up the workstation.

Press the start/stop switch in until it catches in the Start position. The start/stop switch is on the lower right corner of the workstation's front panel (see Figure 2-3). The green light on the switch should go on.

When you press the start/stop switch, the workstation goes through a number of ROM diagnostic tests before you see anything on the console screen. These diagnostic tests are loaded from the workstation's read-only memory (ROM). As each test is executed its test number is displayed on the seven-segment Computer Board Diagnostic LED, located on the workstation's back panel (see Figure 2-1). If a test fails, the test's number remains on the LED. If a failure occurs, see the *6130 System Diagnostics* manual or contact your Tektronix Field Office. If all tests complete successfully, the seven-segment LED momentarily turns off.

At this point, the workstation automatically runs power-up diagnostics. These diagnostics are part of the diagnostic operating system. They are loaded from the Winchester disk and executed. The seven segments of the LED flash in a race track pattern.

Diagnostic messages appear on the console screen. Nonfatal errors also cause messages to the console screen. If a fatal hardware error occurs during these tests, the LED panel on the back of the workstation lights in a pattern that indicates the error. If an error occurs, see the *6130 System Diagnostics* manual or contact your Tektronix Field Office.

During these tests, if the settings of the configuration switches have been changed since the last test start-up at the factory, the following message appears:

System configuration has changed since last boot
Update config file to reflect new configuration? [y,n,(y)]

Answer y to this question. If you just press <RETURN>, the answer defaults to "yes."

After you answer this question, the diagnostic tests start over again and run all the way through.

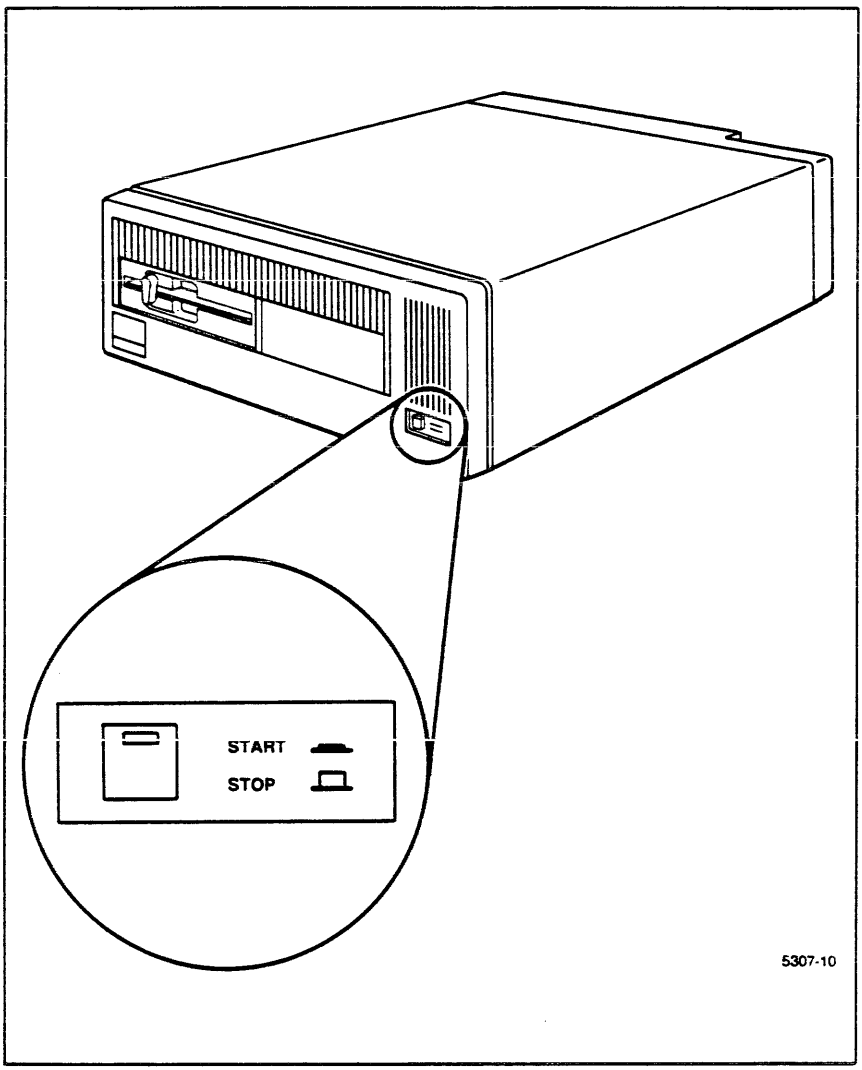


Figure 2-3. Front of the 6130 Workstation

When the diagnostic operating system is finished running tests, the LED momentarily turns off.

If no errors occur during the power-up diagnostics, the workstation boot program begins. Messages from the boot program appear on the screen. These messages include a list of the devices the boot program finds, memory allocation data, and so on. Example 2-1 shows a typical list of these boot messages.

```
<UTek> Tektronix - UTek Tek6130_R3.0 #1.25 Tue May 10 17:39:41 PDT 1988
<UTek> Configuring device drivers:
<UTek>  main: tty, scsi, lan
<UTek> Configuring paging space: 9 megabytes on 1 drive
<UTek>  initial paging space: 9 megabytes on /dev/ds00b
<UTek> end configure
<UTek> Configuring memory: 1 megabytes physical, 16 megabytes virtual
<UTek>  process virtual memory size: 15 megabytes  (1 kilobyte page size)
<UTek>  disk buffers: 32 buffers, 64 kilobytes of physical memory
<UTek>  available for processes: 334 kilobytes of physical memory
```

Example 2-1. Sample Boot Program Messages.

The boot program then checks to see if the */fastboot* file exists. If the file exists, then the file system check (**fsck**) performed during factory testing ended with a healthy file system, so the system does not run **fsck** now.

If the system doesn't run **fsck** at boot, this message appears on the console screen:

```
Fast boot ... skipping disk checks
```

If the boot program doesn't find the */fastboot* file, it starts up **fsck** in *preen* mode. You can tell that the system is running **fsck** if the following message appears on the console screen:

```
Automatic reboot in progress... date
```

where *date* is the date and time according to the workstation's internal clock. Since you have not set this clock yet, the date and time may not be accurate.

If the system runs **fsck**, the boot procedure takes a few minutes longer. There are three possible outcomes for this **fsck**:

- Fsck completes with no errors. A message resembling the following appears on the screen:

  ```
  /dev/dw00a: 1032 files, 9551 used, 3560 free (168 frags, 424 blocks)
  ```

  Then, the boot procedure continues with the *network configuration* program, **netconfig**. Network configuration is discussed next in this section.

- Fsck finds file system errors that it is able to correct. You see the errors reported on the console screen. **Fsck** then signals the system to reboot. The reboot occurs, **fsck** runs again (this time without errors), and the system ends up ready for network configuration.

- Fsck finds file system errors that it cannot correct. This outcome is very rare. The fsck signals the workstation to come up in single-user mode. You know that the workstation is in single-user mode by the number sign prompt, #. If you get this prompt, run **fsck** manually and correct the file system errors. See File System Maintenance in Section 7, and the list of **fsck** messages in Appendix A of this manual for details on running **fsck**.

  Once you have corrected the file system, reboot the system to incorporate the changes made to the file system with **fsck**. There are two ways to reboot the system:

  1. Turn the power off and then on again.
  2. Type

     /etc/reboot

  Either of these methods runs the boot program again. You should end up ready to perform network configuration for the workstation.

NETWORK CONFIGURATION

At this point, the system prompts you to set up the network parameters for your workstation. Your workstation can be connected to a Local Area Network (LAN). For the workstation to communicate properly with other computers on the LAN, certain parameters must be assigned.

You should assign a *hostname* even if you have no plans to attach the workstation to a LAN. A hostname is the name by which other users on a network can address your workstation. If you do not assign a hostname for your workstation, the **login:** prompt for your system is **<No host name set> login:**.

If you are connecting your workstation to a new network that also contains other Tektronix workstations, you should contact the people administering the other workstations and choose someone to administer the network. *Network administrator* responsibilities are discussed in Section 3, LAN Administration.

If you are attaching your workstation to an already-existing network, there should already be a network administrator. Although this person may not be knowledgeable about networking details for Tektronix 6130 workstations, she or he should be able to tell you:

- Whether the hostname you chose is unique.
- What network number you should assign to your workstation.
- Whether the default Internet address for your workstation is unique.

If the network administrator is familiar with 6130 networking, she or he should also tell you what range of *userids* and *groupids* you can use when you add users and groups to your workstation. Network Administration in Section 3 and Users and Groups in Section 6 cover ranges of IDs and why they are necessary.

Setting Network Parameters

The system runs **netconfig** the first time you boot the system. **Netconfig** asks you questions, and the way you answer the questions determines the settings of the network parameters. **Netconfig** keeps track of the following network parameters:

- Hostname
- State of the Network File System (active or inactive)
- State of the standard network utilities (active or inactive)
- Internet address for each network interface (one interface is standard on a 6130 workstation)

If you want to set up your workstation on a network now, refer to the discussion of **netconfig** in Section 3. Otherwise, follow the directions in this discussion as you go through the first time start-up **netconfig**. This procedure sets a hostname for your workstation, but does not set an Internet address or enable either the network file system or the standard network utilities.

The following discussion lists the questions that **netconfig** asks and explains how to answer these questions to get through **netconfig** as quickly as possible.

1. First, the **netconfig** program asks you to set a hostname.

```
Hostname can be up to 32 characters long.
The first character must be alphabetic.
Legal characters are:
       [A-Z,a-z,0-9,-,_]
Enter hostname [string] . . .
```

The hostname must be an alphanumeric string 1 to 32 characters long. The hyphen (-) and underscore (_) characters are also allowed. The first character of the name must be a letter. Some example hostnames are:

charlie

engr1

station_1a

myworkstation

This name should be unique for the entire set of machines that your workstation can talk to. If you are on a small network, or creating a new network, it's not hard to determine if the name you chose is unique.

If, however, you are connected to a *gateway node* that is connected to many other networks, it may be harder to find a unique name. A gateway node is a machine on a LAN that can communicate with two or more LANs, thereby acting as a connection between them. Check the name you choose with the administrator of the gateway node before assigning it to the workstation, since the administrator probably knows names already in use on existing LANs.

If you are assigning a hostname, but not attaching to a network at this point, assign any name you choose that fits the rules just given.

2. When you enter a hostname, the program responds with:

```
New hostname is 'hostname'.
Is that acceptable? [y,n,q(n)] . . .
```

Hostname is the hostname that you just entered. If you want to use the name you just entered, type **y**.

If you want to choose another name, type **n**, and the request to enter a name is repeated.

Do not choose the quit option (**q**) at this point.

3. Once you have set and accepted a hostname, the program asks:

```
Do you wish to enable the Distributed File System?
[y,n,q(n)] . . .
```

Answer **n** to this question. If you *do* want to enable the Network File System, answer **y** here and refer to Section 3 for more information.

Do not use the quit (**q**) option at this point.

4. When you have answered the network file system question, the program asks:

```
Do you want to enable the regular Network Utilities?
[y,n,q(n)] . . .
```

Answer **n** to this question. If you want to enable the standard network utilities (that is, **rsh**, **rlogin**, and **rcp**), answer **y** here and refer to Section 3 for more information.

Do not use the quit (**q**) option at this point.

After you go through the **netconfig** program and give the workstation a hostname, the boot program finishes its tasks and the **login:** prompt appears on the console. The **login:** prompt should look like:

hostname **login:**

where *hostname* is the name you just assigned with the **netconfig** program.

LOGGING IN

Once you have the **login:** prompt on the screen, you can log in for the first time.

NOTE

If you didn't assign a hostname to the workstation with the **netconfig** *program, the* **login:** *prompt says* **<No host name set> login:**. *The prompt remains like this until you assign a hostname.*

Some accounts already exist when the system is first brought up. You use these accounts to perform system administration tasks.

Table 2-3 lists these accounts.

Table 2-3
ORIGINAL 6130 ACCOUNTS

Account Name	Purpose
root	Root account
sysadmin	Sysadmin interface
daemon	Owner of some system daemons
cron	Owner of cron daemon
sys	Owner of some system files
dist	Owner of distribution tapes
uucp	Owner of uucp communications
admin	Administrator
mdqs	Multidevice queuing system account
dumpopr	Fileserver dumping account
dumprmt	Remote fileserver dumping account
user	General user account
test	Test user account

The first account you should log into is the *root* account.

Log in by typing:

root

You then get a message telling you when the last time someone logged in as *root* (at the factory), and the message:

TERM = (4205)

This **TERM** line is a prompt that identifies the type of terminal the system expects. UTek treats different terminals in slightly different ways to provide correct screen representation of characters, correct visual editor (**vi**) representation, and so on.

UTek recognizes many terminal types. If you do not know the name by which UTek knows your terminal, see Appendix D for a list of the most common terminal names. If your terminal is not in this list, contact your local Tektronix Field Office for this information. A discussion of termcaps is available in *termcap(5T)*.

If the terminal you are using as console is a Tektronix 4205 Computer Display Terminal, enter <RETURN> in response to this prompt. Otherwise, enter the two to five letter representation of the terminal you are using that UTek recognizes, and then press <RETURN>. UTek tries to match the string you enter with the entries in the file */etc/termcap*. For example, if you have an Tektronix 4207, instead of a 4205, enter **4207**.

Once you have responded to the **TERM** prompt, you get the root command prompt, a # (number sign). This is the prompt that tells you the system is ready to accept a command. The root prompt is different than the prompts for regular user accounts so that you know that you are *root* with special root privileges.

Root Account

The root account is the most powerful account on the system. It lets you access any file on the system, regardless of the who owns the file or how the file's permissions are set.

CAUTION

When you are logged into the root account, you can destroy the system by changing or removing vital system files. You should only log into the root account when you need to perform system administration tasks that cannot be performed from any other account. Use a regular user account for your personal work, and the sysadmin *account for system administration tasks whenever possible.*

When you are in the root account, you are known as *root* or *superuser*. Along with your powers as root comes great responsibility. Do not abuse your superuser powers by manipulating the system or other users' files. Before you access, create, or destroy any file or process owned by a user, ask the user's permission.

When you are creating user accounts, be sure to create an account for yourself so that you have somewhere besides the root account to do your work. If you work in the root account, you may destroy the system accidentally.

Because of the power of the root account, the password for the account should be kept secure. However, there should be at least two people who know the password so that system administration tasks can be done even if the regular system administrator (you) is not available. Section 7 discusses system security.

Setting Passwords

Your first activity as root should be to assign passwords to the accounts in Table 2-3. None of these accounts have passwords yet and, since these accounts include the most powerful accounts on the system, you should set passwords for all these accounts *immediately*.

Use the **passwd** command to assign a password for each of these accounts. Passwords must be a minimum of five characters and a maximum of eight characters long.

Passwords for these special accounts must be especially secure. Using six or more characters, and using non-alphanumeric characters, are good ways to ensure greater security.

To assign a password to the root account:

1. Type:

   passwd root

   This begins the following exchange:

   ```
   Changing password for root
   New password:
   Retype new password:
   ```

2. Enter the password you choose in response to the **New password:** prompt, press <RETURN>, then repeat it in response to the **Retype new password:** prompt. The password you enter does not show on the screen for security purposes.

Repeat the above procedure for all the accounts in Table 2-3. Specify the account name after the **passwd** command. You can remain in the *root* account to set the passwords for these or any other accounts. If you are in an account other than *root*, you can set only the password for that account.

You may want to leave the *user* account without a password, so that any user may log in and use the system without having an account. In other words, the *user* account would be a public account.

Logging Out of the root Account

When you are finished working in the root account, leave by pressing <CTRL-D>. The **login:** prompt reappears on the screen.

INSTALLING OPTIONAL SOFTWARE

The software already installed on the Winchester disk of your 6130 workstation includes the core UTek package. If you have purchased any optional software packages, you must install them onto the Winchester disk before you can use them.

Optional UTek software for the Tektronix 6130 workstation includes:

- UTek/A, the Auxiliary Utilities package (64WP05).
- UTek/PS, the Programming Support package (64WP06).
- Programming languages. Available are C, FORTRAN 77, BASIC, and Pascal.

There are also Tektronix applications that are tailored to the 6130 workstation, and software to support future hardware options will become available.

You can install optional software with the Install Optional Software option to the *sysadmin interface*. See the Install Optional Software discussion in Section 4 and the installation instructions that come with the software package for details on installing optional software.

Be sure to take a system backup after you have installed all optional software. See the discussions on Backups in Section 4.

SETTING THE DATE AND TIME

The workstation's clock must be properly set, because often users assign certain times for tasks to occur (using the at command). Also, you can set up system backups or other system tasks that rely on low system use to occur automatically. If the clock is incorrect, these jobs can start running at unexpected times, using system resources needed elsewhere.

The workstation clock should only need setting once, when you start the system for the first time. However, you should reset the clock if the workstation is moved to a new time zone, or if for some reason the clock stopped (such as the computer board was changed). If you want the workstation to reflect a new time zone, or daylight/standard time changes, you can use the *sysconf* utility to permanently set these (see Section 9 for details).

The workstation's start-up procedure prompts you to check and correct the clock when more than 24 hours have passed since the workstation last had power. It does this by asking you to check the date.

To set the date and the clock, use the **date** command. You must be logged in as *root* to execute the **date** command. The syntax for this command is (information inside brackets is optional):

date -z *timezone [yy]mmddhhmm[.ss]*

where:

-z *timezone* is the time zone you are in. Table 2-4 shows valid time zone specifiers and the zones they represent.

yy is a two-digit field representing *year*. For example, 84 would indicate 1984. If you don't include this field, the last year entered is assumed.

mm is a two-digit field representing *month*. For example, 03 would indicate March.

dd is a two-digit field representing *day*. For example, 12 indicates the twelfth day of the month.

hh is a two-digit field representing *hours*. The 24-hour clock is used.

mm is a two-digit field representing *minutes*.

.ss is a two-digit field preceded by a period (.) representing *seconds*. If you omit this field, the clock starts counting at .00 seconds of the minute you set.

For more information on setting the time and date see, *date(1)*.

Table 2-4
TIME ZONE SPECIFIERS

Specifier (Standard Time)	Specifier (Daylight Time)	Zone
EET	EET	Eastern European
MET	MET	Middle European
WET	WET	Western European
AST	ADT	Atlantic
EST	EDT	Eastern
CST	CDT	Central
MST	MDT	Mountain
PST	PDT	Pacific
AEST	AEST	Eastern Australian
ACST	ACST	Central Australian
AWST	AWST	Western Australian

SOFTWARE INSTALLATION OF EXTERNAL PERIPHERALS

All devices (terminals, disk drives, printers, and so on) that are on the system must have files in the */dev* directory. Then, when you want to access a device, your program addresses the file in */dev* that corresponds to that device.

When you first bring up the workstation, a number of these device files already exist in */dev*. Table 2-5 shows the names of these device files, and which devices they represent.

Table 2-5
STANDARD DEVICES

Device File	Device
/dev/dw00a -- /dev/dw00p	Winchester disk partitions
/dev/rdw00a -- /dev/rdw00p	Raw mode Winchester disk partitions
/dev/df	Diskette drive
/dev/rdf	Raw diskette drive
/dev/tty00	RS-232-C port 0
/dev/tty01	RS-232-C port 1
/dev/gpib0	GPIB port
/dev/gpid0	GPIB configuration device
/dev/ttyp0 -- /dev/ttyp3	Pseudodevices for remote logins
/dev/ptyp0 -- /dev/ptyp3	Pseudodevices for network software
/dev/mem, /dev/kmem, /dev/cvt, /dev/drum	Physical memory devices
/dev/console	Console device
/dev/tty	Current terminal
/dev/null	Null device for programming

You can also add other devices if a *driver* for them exists in the kernel. Devices that are not listed in the standard kernel come with a diskette in the sysconfig package that contains the driver. You load this diskette and install the driver in the kernel using the *sysadmin* interface (see Section 4, Installing Optional Software). You can then create device files with the **MAKEDEV** utility for Tektronix peripherals you add. Section 5 discusses creating these new device files under Adding Devices. You must be logged in as *root* to create device files in */dev*.

If you have enhancements for a printer or streaming cartridge tape drive (Dual Hardcopy Interface or SCSI enhancements), you must create device files before you use peripherals attached to the interfaces. You can create the file later (see Section 5), but you cannot use the peripheral until the file exists.

Also, if you install a printer, you should set up a print queue with the Multidevice Queuing System (MDQS). You can create the queue with the sysadmin interface. More information on the MDQS is available in Section 4 and in the *UTek Tools* book.

WHAT NOW?

When you reach this point, there are two things that you should do. The order in which you do them depends upon your knowledge of the UTek system and upon the speed at which your workstation must become available to users.

The two things are:

1. Learn more about the UTek system by reading parts of the documentation as specified in the following paragraphs.
2. Set up user accounts for all the people who use the system.

More About UTek

You should prepare yourself for the task of system administration. There are a number of documents in the UTek system documentation set that you should read, depending on your current level of knowledge.

If you've never used a UNIX-based system before, read the *6130 Learning Guide* and do the UTek online sessions. Also, read at least Chapter 14 of *Introducing the UNIX System* and the sections UTek System Implementation, UTek Fast File System, and the Distributed File System in the *UTek Tools* book.

You should also learn how to use one of the UTek text editors. Information on text editors is available in *Introducing the UNIX System* and in the *UTek Tools* book.

If you've used a UNIX-based system before, but have never been a system administrator, read Chapter 14 of *Introducing the UNIX System*, and the UTek System Implementation, UTek Fast File System, and the Distributed File System sections in the *UTek Tools* book.

If you've been system administrator for a UNIX-based system before, read UTek System Implementation, UTek Fast File System, and the Distributed File System in the *UTek Tools* book. These documents cover features of UTek that you may not have encountered before.

for more information about how the Network File System (NFS) works, read the *Netwrok File System Reference* manual.

Further System Administration Tasks

As system administrator, you must perform these tasks before users can log on to the system:

- Set up user accounts.
- Set up groups.
- Set up queues for any peripheral devices.
- Configure the system to allow mail between machines.
- Configure the RS-232-C ports for the devices attached to them.

These tasks can be done with the sysadmin interface. For details, see Section 4 of this manual.

3

LAN Administration

INTRODUCTION

This section describes the tasks you must perform and information you must know as the system administrator on a workstation connected to a local area network (LAN). Described in this section are:

- How to configure the network software after connecting your workstation to a network.
- How to control who on your network can access your workstation.
- How to assign the proper user identification numbers and group identification numbers so that network security is preserved.
- How to create and maintain the files related to networking on your workstation.
- How to find out what is happening on your network.
- How to configure your workstation as a file server for other workstations on the network.

This section also describes the tasks you must perform if you become the *network administrator*, the person in charge of the entire network.

Using the LAN and sending mail over the LAN are not described in this section. For information on using the LAN, see the Local Area Network section of the *6130 System User's Guide*. For information on sending mail to other hosts on the LAN, see the Sendmail Configuration topic in Section 4 and the information on mail in the *UTek Tools* manual.

Before reading this section, you should:

- If you are creating a new network, install the LAN cable.
- Set up your LAN transceiver. See the instructions that came with it.
- Try to get a list of the names of all the workstations and computers on your network, along with their Internet addresses. This list could be quite long, but it is necessary to help you choose a unique name and Internet address for your workstation.
- Decide on a name to identify your workstation on the network.
- Determine the class (A, B, or C) of Internet addressing that is to be used on your network. If you are connecting to an existing network, ask your network administrator. If you are creating a new network, decide on the class you want to use. Internet address classes are discussed later in this section.

WHAT IS A LAN?

A *local area network* is a linking together of workstations and computers that lets users access more than one host on the LAN. Each workstation and computer connected to the LAN is called a *node* or *host* of that network.

Each node is connected to a large coaxial cable by devices called *transceivers*, which assure that each node on the network transmits and receives data properly. Each 6130 workstation has one LAN interface installed, designated *lna0*. The LAN interface can be accessed through a port on the back panel of the workstation. Figure 3-1 shows the hardware components of a typical local area network.

Each node on the LAN has two unique addresses: an *Internet address* and an *Ethernet address*. The network software uses a host's Internet address when generating messages to send to another host. Before these messages are sent out over the network, the Internet addresses are converted to Ethernet addresses, which the network hardware uses to communicate with the network. Ethernet addresses are assigned at the factory by Tektronix.

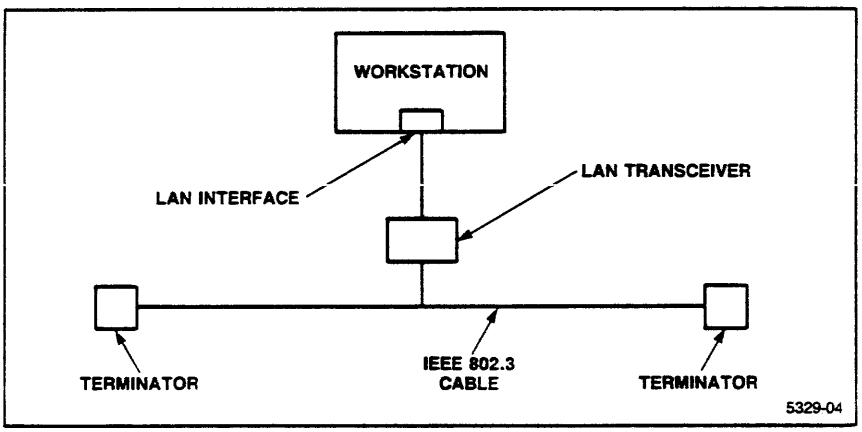


Figure 3-1. Local Area Network Components

Each host on the LAN also has a unique *hostname*, which can be used instead of the Internet address to generate messages to send over the network. As a system administrator, you must select an Internet address and a hostname for your workstation. In unusual circumstances you may also have to reassign the Ethernet address. See the *6130 Diagnostics* manual if you think you need to change the Ethernet address.

A node that is connected to two or more different LANs is called a *gateway node* (Figure 3-2). A gateway node lets users on one network access hosts on another network.

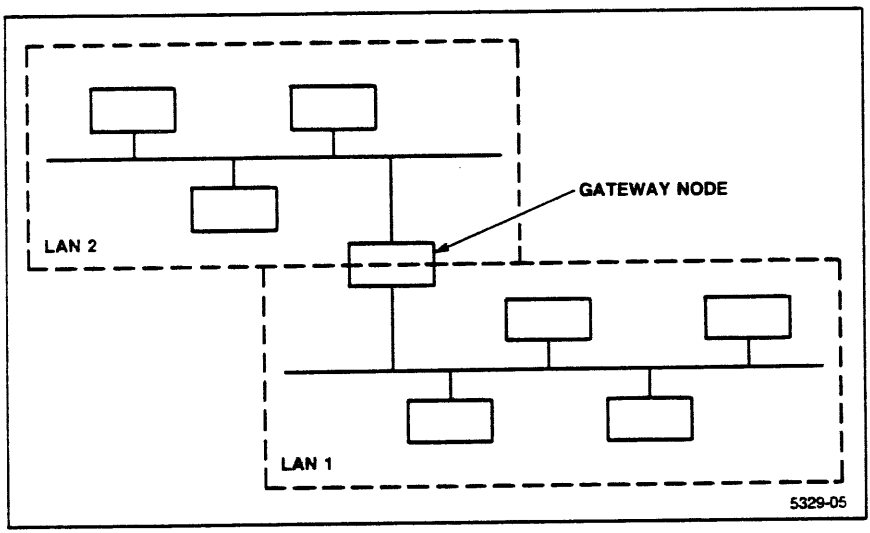


Figure 3-2. Gateway Node

NETWORK SOFTWARE MODEL

The network software on your workstation is based on the International Standards Organization's seven-layer Open Systems Interconnection model. Figure 3-3 shows the seven layers of the ISO model. The top three layers (the *applications*, *presentation*, and *session* layers) are combined into one layer, labeled the *application program* layer.

Data sent over the network must pass from the *application program* layer to the *transport* layer to the *network* layer to the *link* layer and finally to the *physical* layer. Data received from the network travels in reverse: first through the *physical layer* to the *link layer* to the *network layer* to the *transport layer* and finally up to the *application program* layer.

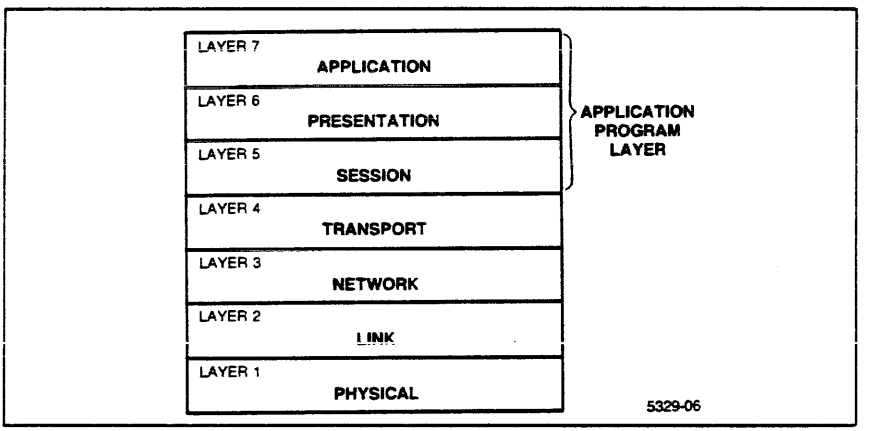


Figure 3-3. Layers Defined by the ISO Model

Table 3-1 shows how some of the software on your workstation fits into the ISO model.

Table 3-1 shows how one layer uses the layers below it. For example, the **rcp** (remote copy) program uses the TCP, IP, IEEE std 802.2 Controller Software, and the IEEE std 802.3 (Ethernet) hardware layers.

When you use the **rcp** command to copy a file to a remote host, **rcp** passes the data from the file to be transferred to TCP, which puts the data into *packets* (units of data) that can be sent over the network. TCP passes these packets to IP, which finds the route the packets need to travel to reach the remote host. IP passes the data to the link layer, which encapsulates the data in an Ethernet frame and interacts with the IEEE std 802.3 hardware to send the frame on the network media.

Table 3-1
NETWORK SOFTWARE AND ISO LAYERS

Layer Number	ISO Layer	Software Component		
			Mail	
5	Applications, Presentation, and Session	FTP TELNET rcp rlogin rsh	uucp SMTP	nameserver netstat who
4	Transport	TCP		UDP
3	Network	IP		
2	Link	IEEE std 802.2 Controller Software		
1	Physical	Ethernet		

CONFIGURING THE NETWORK SOFTWARE

After you have physically connected your workstation to a network, take these steps to configure the network software for your network. These steps are explained in more detail in the remainder of this chapter.

1. Run **netconfig** or **sysadmin** to set the hostname and Internet address of your workstation. If you set the hostname and Internet address when you first started up the workstation, you don't need to do it here, unless there is a conflict with other hosts on the network.

   If you have other tasks to do using **sysadmin**, you can follow the procedure described in Section 4 for setting the Internet address and hostname with **sysadmin**. Otherwise, you can follow the procedure in this section for using **netconfig**.

2. Run **netconfig** to signal your workstation to start up the *network daemons* the next time your system goes from single-user to multiuser mode. See the Running Netconfig topic in this section. For more on daemons and what they are, see the discussion on daemon processes in Section 6.

3. Edit the network files that control the network software. Some files you may need to edit are */etc/hosts* and */etc/hosts.equiv*. See the description of the */etc/hosts.equiv* file later in this section for more information or read the appropriate pages in Section 5N of the *UTek Command Reference* manual.

4. If your workstation is in multiuser mode, bring the workstation down to single-user mode and back up to multiuser mode with the **/etc/shutdown** command for the changes in steps 1, 2, and 3 to take effect. See Section 5 for more information on **/etc/shutdown**.

5. Have the system administrators of the other hosts on your network include your workstation in their network files, so that you can access those hosts.

6. Configure the mail system of your workstation so you can send mail to other hosts on your network. See the Mail Routing Configuration heading in Section 4 and *sendmail(1)* in the *UTek Command Reference* manual.

RUNNING NETCONFIG

Netconfig is the program that configures the network software of your workstation. **Netconfig** is used to:

- Set the hostname of your workstation.
- Tell your system to enable or disable the network file system the next time the workstation goes from single-user to multiuser mode
- Tell your system to turn the network daemons (which let you use the regular network utilities) off or on the next time the workstation goes from single-user to multiuser mode.
- Set the Internet address of the network interface(s).

Netconfig runs automatically the first time you start up your workstation. If you don't enter valid responses to the requests of **netconfig**, then **netconfig** is run automatically every time you boot (start up) your workstation.

You can also run **netconfig** by logging in as *root* and then typing:

```
/etc/netconfig
```

The following paragraphs describe the information **netconfig** prompts you for when you bring your workstation up or when you type **/etc/netconfig** and what you should enter.

Throughout **netconfig** you are prompted to answer a question that is in the following format:

Question `[y, n, q (n)] . . .`

Question can be answered by entering **y** (meaning *yes*), **n** (meaning *no*), or **q** (meaning *quit* the program). If you press <RETURN> without typing any of the letters or if you enter **q** to quit, you get the default choice, which is the letter in parentheses.

The changes you make with **netconfig** don't take effect until the next time your workstation goes from single-user mode to multiuser mode. You can cause the workstation to go from single-user mode to multiuser mode by entering the **/etc/shutdown** command to bring the workstation down to single-user mode and then pressing <CTRL-D> to bring the workstation back up to multiuser mode. The workstation also goes from single-user mode to multiuser mode when it is booted. See Section 5 for more information.

Setting the Hostname

The first thing **netconfig** prompts you for is a hostname. The *hostname* is the name by which other users on the network can address your workstation. If your workstation already has a hostname, **netconfig** asks you:

```
The current hostname is "hostname".
Is that acceptable? [y, n, q (n)] . . .
```

If there is no conflict between your hostname and other hosts on your network and if this name is what you want, enter **y** (meaning *yes*).

If the current hostname is not acceptable, enter **n** (meaning *no*) and then enter a new hostname when **netconfig** prompts you with the following line. This is also the prompt you see if you are setting the hostname for the first time.

```
Enter hostname [string] . . .
```

The hostname must be an alphanumeric string 1 to 32 characters long. The hyphen (-) and underscore (_) characters are also allowed. The first character of the name must be a letter. Some example hostnames are:

charlie
engr1
station_1a

This name should be different from the names of the hosts that your workstation can talk to over the LAN. If one person is in charge of your network, check the name you chose with that person before assigning it to your workstation.

You can determine the current hostname of your workstation by entering the **hostname** command or by reading the name that appears in your login prompt. If the hostname of your workstation hasn't been set, the prompt appears as:

```
<No host name set> login:
```

Controlling Network Daemons

Then **netconfig** asks you:

```
Do you want to enable the regular Network Utilities?
[y,n,q(n)]...
```

Enter **y** if you want to use the **rcp**, **rsh**, and **rlogin** commands on your workstation. Entering **y** enables the network daemons for these commands the next time your workstation goes from single-user mode to multiuser mode. Enter **n** if you want to disable the network utilities.

If you enter **y** and you haven't already set the Internet address of your workstation's network interface, you are prompted to enter the Internet address.

Setting the Internet Address

If you entered **y** to the question about enabling the network file system or the regular network utilities, and you have set the Internet address before, **netconfig** asks you to verify the Internet address for the workstation's network interface(s). Remember the name of the LAN interface is *lna0*. For example:

```
The internet address for lna0 including network number is
      7.0.80.9
Is that acceptable [y, n, or q] . . .
```

Check this Internet address against the list of Internet addresses already in use on your network. Enter **y** if this address is unique on your network and skip to the Controlling Network Daemons topic later in this section. Otherwise, enter **n** and you are asked:

```
Enter the network number portion of the address:. . .
```

The *network number* is a portion of the Internet address. This number identifies your network and the class (A, B, or C) of Internet addressing you are using. You must enter the Internet address in the correct format for the class of addressing used on your network. Internet address classes are described in more detail in the topic Ethernet and Internet Addresses later in this section.

Table 3-2 summarizes how to enter the network number portion of your Internet address, depending on the class of addressing you are using.

NOTE

Network number 127 (class A) is reserved and used to indicate the loopback network. A loopback network is a network that has only the local host (your workstation) on it. The full loopback internet address used on 6130 workstations is 127.0.0.1. This address always refers to the local host, no matter which host you are on. The loopback capability is primarily used for diagnostics.

If you are using Class A addressing, enter the network number as an integer in the range 0-126. This number represents the most-significant 8 bits (bits 0-7) of your Internet address.

Table 3-2
ENTERING A NETWORK NUMBER

Class	Enter	Where
A	x	$0 < x \leq 126$
B	$x.y$	$128 \leq x \leq 191$, $0 \leq y < 256$
C	$x.y.z$	$192 \leq x \leq 223$ $0 \leq y < 256$ $0 \leq z < 256$

If you are using Class B addressing, enter the network number as two integers separated by periods. The first integer represents the most-significant 8 bits (bits 0-7) of your Internet address and must be in the range 128-191. The second integer represents the next 8 bits (bits 8-15) of your Internet address.

If you are using Class C addressing, enter the network number as three integers, separated by periods. The first integer represents the most-significant 8 bits (bits 0-7) of your Internet address and must be in the range 192-223. The second integer represents the next 8 bits (bits 8-15) of your Internet address and the third integer represents the next 8 bits (bits 16-23) of your Internet address.

If you are creating a new network that doesn't contain a gateway node to other networks, you can use any class of addressing. Be sure you assign the same network number to all the workstations you are connecting to the new network.

If you are attaching your workstation to an existing network (or if you are going to connect to an existing network in the future), get the network number from the network administrator. If you are creating a new network containing a gateway node, get the network number from your network administrator.

After you enter the network number, **netconfig** creates a default Internet address by appending 8, 16, or 24 bits from your workstation's Ethernet address to the network number to form a 32-bit address. Then **netconfig** prompts you to determine if the default Internet address is unique. For example:

```
An address based on the ethernet address for lna0 is:
    7.23.4.12
Is that acceptable? [y, n, q (n)] . . .
```

If this address is unique on your network, enter **y**. If this address is not unique, enter **n**. If you enter **n**, **netconfig** prompts you for the network number and the rest of the address needed to construct an Internet address. The rest of the address is called the *host address*.

Table 3-3 summarizes what you should enter for the host address after entering the network number, based on the class of addressing you are using:

Table 3-3
ENTERING A HOST ADDRESS

Class	Enter	Where
A	*x.y.z*	$0 \leq x, y, z \leq 255$
B	*x.y*	$0 \leq x, y \leq 255$
C	*x*	$0 \leq x \leq 255$

An example of setting the Internet address might look like this:

```
An address based on the ethernet address for lna0 is:
    129.23.4.12
Is that acceptable? [y, n, q (n)] . . .n
Enter the network number portion of address:. . .129.23
Enter the address (ARPA format is ddd.ddd.ddd) . . .8.123
The internet address for lna0 including network portion is:
    129.23.8.123
Is that acceptable [y, n, q (n)] . . . y
The internet address for lna0 has been set.
```

Other Netconfig Options

Netconfig has some other options that you may use. For example:

/etc/netconfig –P	Prints the Ethernet and Internet addresses of your workstation.
/etc/netconfig –e net	Causes the network daemons to be enabled the next time the workstation goes from single-user to multiuser mode.
/etc/netconfig –d net	Causes the network daemons to be disabled the next time the workstation goes from single-user to multiuser mode.

See *netconfig(8N)* in the *UTek Command Reference* manual for more information on netconfig and its various options.

NOTE

Remember, any changes you make with **netconfig** *have no effect until the file* /etc/rc.net *is run. To run this file, use the command* **/etc/rc.net** *when you are logged in as* root, *or bring the system to single user mode and then back up to multiuser mode.*

The changes you make with **netconfig** don't take effect until the next time your workstation goes from single-user mode to multiuser mode. You can cause the workstation to go from single-user mode to multiuser mode by entering the **/etc/shutdown** command to bring the workstation down to single-user mode and then pressing <CTRL-D> to bring the workstation back up to multiuser mode. The workstation also goes from single-user mode to multiuser mode when it is booted. See Section 5 for more information.

TYPES OF LAN ACCESS

The two most common ways that your workstation can share files with other nodes on the LAN are:

- The remote commands **rlogin**, **rsh**, and **rcp**
- The network file system (NFS)

The hosts you can access with each of these file-sharing methods is called the *domain* of that method. Of these file-sharing methods each has different files that control its domain. As the system administrator, you must specify the domains by editing these files on your workstation.

Remote Commands

The commands that let you access files on other workstations on your LAN are called *remote commands*. There are three remote commands:

rcp (Remote copy) Copies files between two hosts on the LAN.

rlogin (Remote login) Logs you into another host on the LAN.

rsh (Remote shell) Executes a single command on another host on the LAN.

Figure 3-4 shows how the remote commands work. Note that when you use a remote command, it is as if your terminal is directly connected to the remote host during the time the command is running. These commands let you switch between your host and any other host on the network that has the proper files set up.

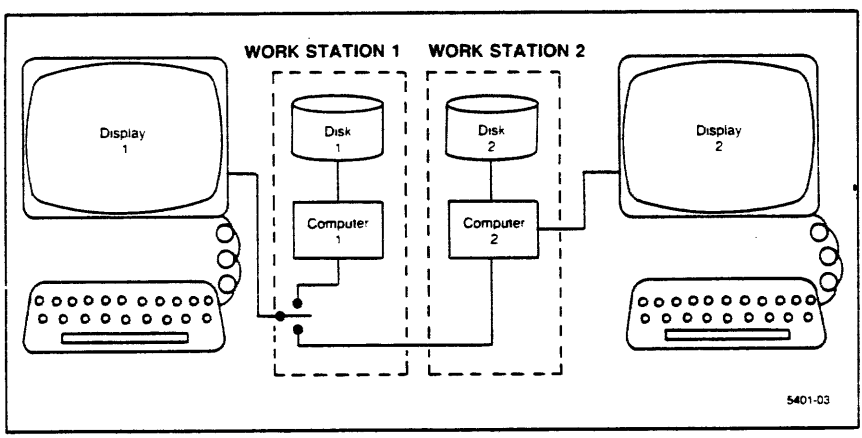


Figure 3-4. How Remote Commands Work.

As the system administrator, you control who can access your workstation with these remote commands by putting entries in the */etc/hosts.equiv* file. Individual users on your workstation can also allow access to their accounts by creating *.rhosts* files in their home directories.

/etc/hosts.equiv

The */etc/hosts.equiv* file contains the names of all the hosts on the network whose users can access your workstation with the **rcp**, **rsh**, and **rlogin** commands.

Each line in the */etc/hosts.equiv* file has the following format:

hostname [*username*]

where *hostname* is the name of a host on your network and *username* is the login name of a user on that host. *Username* can be the login name of any user except *root*, for security reasons. If you omit *username*, all users (except *root*) on the remote host can access your workstation.

The */etc/hosts.equiv* file controls who can access your workstation but, even after a user gains access to your workstation, their access to files on your system is controlled by the protection mode you set on your files.

See *hosts.equiv(5N)* in the *UTek Command Reference* manual for more information.

You can also control access to the **rcp**, **rsh**, and **rlogin** commands with the *.rhosts* file.

Even if you don't list hosts in the */etc/hosts.equiv* file, individual users on your workstation can let remote users access your workstation with the **rcp**, **rsh**, and **rlogin** commands by creating files named *.rhosts* in their home directories.

When a user on another host tries to access your workstation with a remote command, the network software checks to see if that user's host is listed in the */etc/hosts.equiv* file. If not, then the network software checks that person's home directory on your workstation (or the home directory of the user specified by the person trying to access your workstation) looking for a *.rhosts* file. If the *.rhosts* file exists and contains an entry for that user, that user is allowed access.

Lines in the *.rhosts* file have the same format as lines in the */etc/hosts.equiv* file:

hostname [*username*]

where *hostname* is the name of a host on your network and *username* is the name of a user on the remote host. *Username* can be the login name of any user. If *username* is omitted, then only the user whose home directory the *.rhosts* file is in can access your workstation.

For example, if a user named *anne* created a *.rhosts* file in her home directory on *engr2* with the following contents:

```
engr1
engr1 joe
```

Then both *anne* and *joe* could copy files between *engr2* and *engr1* with the **rcp** command while logged into *engr2*. The command line the user logged in to *engr2* as *anne* would enter to remotely copy to *engr2* from *engr1* is:

`rcp engr2:`*fullpath filename*

The command *joe* would use while logged in to *engr1* to **rcp** to *engr2* from *engr1* is:

`rcp engr2.anne:`*fullpath filename*

With the above command, *joe* is given the same file permissions on *engr1* as *anne* has. See *rcp(1)* in the *UTek Command Reference* manual for more information.

Also, the above *.rhosts* file lets *joe* access any files on *engr2* that *anne* normally could using the **rsh** commands from *engr1*. *Joe* can also **rlogin** to *engr2* as *anne* from *engr1* without having to enter *anne*'s password.

For more information on the *.rhosts* file, see *hosts.equiv(5N)* in the *UTek Command Reference* manual. For more information on the remote commands, see *rcp(1)*, *rsh(1)*, and *rlogin(1)*.

ETHERNET AND INTERNET ADDRESSES

Ethernet addresses are used by the network hardware of your workstation, and Internet addresses are used by the higher levels of network software on your workstation.

The LAN hardware on your workstation communicates with the LAN using Ethernet protocols and Ethernet addresses. This makes your workstation compatible with Ethernet hardware from other manufacturers. Ethernet addresses are 48 bits long and are intended to be unique for all computers everywhere.

The higher levels of LAN software on your workstation use 32-bit Internet addresses to communicate. For the higher and lower levels of network software to communicate with each other, the Address Resolution Protocol (arp) maps Internet addresses to Ethernet addresses.

For example, if you use **rsh** to execute a command on *engr1*, the **rsh** command requests the **nameserver** daemon to convert the hostname *engr1* to an Internet address. Then **arp** converts the Internet address of *engr1* to the Ethernet address of *engr1* by looking in the **arp** tables. Then the network hardware uses that Ethernet address to send the **rsh** request to *engr1*.

If **arp** doesn't find the Ethernet address of *engr1* in the **arp** tables, **arp** broadcasts a message over the network requesting the Ethernet address of *engr1*. If **arp** runs on *engr1*, it receives the request and sends the Ethernet address back to your workstation.

If the **nameserver** cannot find an Internet address that corresponds to *engr1*, you receive an *Unknown host* message.

If **arp** isn't running on *engr1*, then you receive a *Connection timed out* message. If you know the Ethernet address of *engr1* is 0207010340C0 (hexadecimal), you can put it in the **arp** tables by typing:

```
/etc/arp -s engr1 2:7:1:3:4Ø:CØ
```

Then next time you try to access *engr1*, **arp** finds the Ethernet address of *engr1*.

You can read more about **arp** in *arp(8N)* and *arp(4N)* in the *UTek Command Reference* manual.

Internet Address Classes

The 32-bit Internet address is separated into a *network number*, the number that uniquely identifies your network, and a *host address*, the number that uniquely identifies your workstation on your network.

The number of bits of the Internet address that are used for the network number determine the *class* of your workstation's Internet address. There are three classes of Internet addresses: *Class A*, *Class B*, and *Class C* addresses.

If your network uses Class A addressing, your network can contain over 16 million nodes. If your network uses Class B addressing, it can contain 65,536 nodes. If your network uses Class C addressing, it can contain 256 nodes.

In Class A addresses, the network number uses one byte (7 bits for the network number, 1 zero bit) of the Internet address. The remaining three bytes (24 bits) of the Internet address form the host number. See Figure 3-6.

In Class B addresses, the network number uses two bytes (14 bits for the network number, 2 bits set to '10') of the Internet address. The remaining two bytes (16 bits) of the Internet address form the host number. See Figure 3-7.

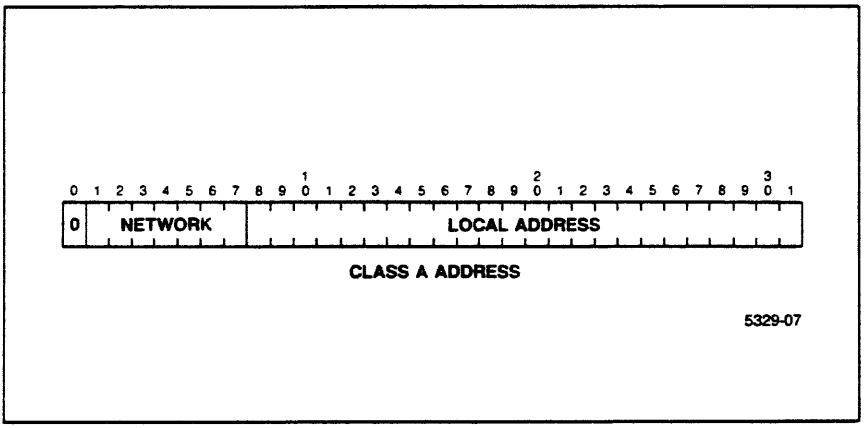


Figure 3–6. Class A Address.

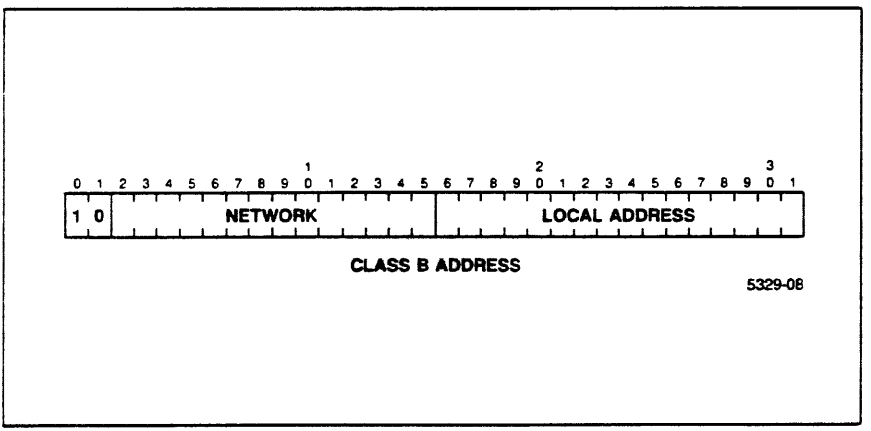


Figure 3–7. Class B Address.

In Class C addresses, the network number uses three bytes (21 bits for the network number, 3 bits set to '110') of the Internet address. The remaining byte (8 bits) of the Internet address form the host number. See Figure 3-8.

Only one of the three classes of Internet addresses can be used on your network, and you must use that type of address on your workstation. If you are connecting your workstation to an existing network, check with the administrators of the other hosts on the network to find which class of address you must use on your workstation.

Table 3-4 is a summary of Internet address classes.

Table 3-4
SUMMARY OF INTERNET ADDRESS CLASSES

Address Type	Network Number		Host Address	
	Size (bits)	Number (networks)	Size (bits)	Number (nodes)
Class A	7	128	24	16,777,216
Class B	14	16,384	16	65,536
Class C	21	2,097,152	8	256

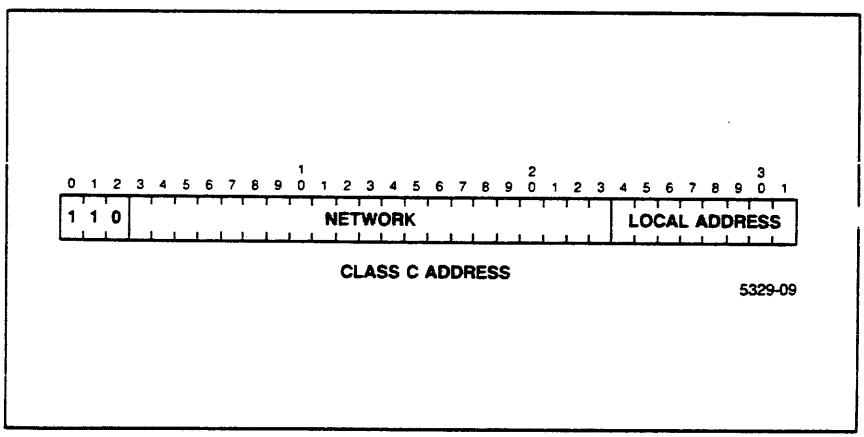


Figure 3-7. Class C Address

ASSIGNING USERIDS AND GROUPIDS

To maintain network security and reduce confusion, every user on your network must have a unique login name and user identification number, called a *userid.* Also, group names and group identification numbers, called *groupids*, must be unique on the network.

For example, if user Joe is assigned a userid of 150 on *host1* and Anne is assigned a userid of 150 on *host2* and both users have access to both hosts, then (with NFS) Joe could access any of Anne's files on *host1* or *host2* and Anne could access any of Joe's files on *host1* or *host2*.

The simplest way to ensure that userids and groupids are unique is to get together with the system administrators of the other hosts on your network and assign a range of userids and groupids to each host. For example, if there are four hosts on a network named *engr1*, *engr2*, *engr3*, and *engr4*, each host could be assigned a range of userids and groupids as follows:

Host	Userids	Groupids
engr1	101-200	101-120
engr2	201-300	121-140
engr3	301-400	141-160
engr4	401-500	161-180

Then the system administrators of these hosts would assign groupids and userids to the users and groups on their workstations from these ranges.

Do not assign userids or groupid that are in the range 0-100. The system uses these numbers for system accounts, such as *root*, *sys*, and others. The highest userid or groupid allowed is 32767.

If a user can use more than one host on the network, the host that the user normally logs in on or the host on which the user first had an account is called their *home machine*. Users of multiple hosts on the network should be assigned a userid from the range assigned to their home machine. Also, users should be listed in the password file (*/etc/passwd*) of every host they use and should have the same userid on all hosts where they have accounts. It is easiest to assign the first userid that a user was assigned to all subsequent accounts for that user. Part of a user's personal information field in the password file entry can specify the user's home machine. See the discussion on personal information and the sysadmin interface's personal information form in Section 4.

If you find that two users on the network have the same userid, you can use the following procedure to change the userid of one of them so there is no longer a conflict. You would probably discover such an error when you try to add a user to your workstation whose userid conflicts with an existing user's. It is easiest to change the userid of the user with accounts on fewer workstations. Remember to change the user's userid on all hosts where the user has an entry in the */etc/passwd* file.

1. Get into the sysadmin interface. (Log in as *sysadmin* or as *root* and type **sysadmin**.)
2. Choose *User Login Account Maintenance* on the top-level System Administration menu.
3. Choose *Change User Information or Add User* on the User Account Maintenance menu.
4. Enter the user's name in the Login name field to call up the user's information.
5. Select the *Edit Account Information* menu item from the User Accounting Change/Add Menu.
6. Remember the old userid, you'll need it in the next step. Change the user's userid to a new, unique userid in the proper range and press <RETURN>.
7. Press the <ESC> key twice. The first press takes you to the User Accounting Change/Add menu. The second press takes you back to the User Accounting Maintenance menu.
8. Save the change you made by pressing **s**.
9. Leave the sysadmin interface by pressing **q**..
10. Type:

    cd /

    This moves you to the root directory.
11. Type in the following command to change the ownership security of the files to the new userid.

    find / -user *olduserid* **-exec /etc/chown** *username* **{} ;**

    where *olduserid* is the userid you just changed and *username* is the login name of the user whose userid you are changing.

NETWORK FILES

A number of files control how the network software works. The */etc/hosts.equiv*, *.rhosts*, and */etc/dfs.access* files were described earlier in this section. Other files used by the network software are described in the following paragraphs.

/etc/hosts

The */etc/hosts* file contains the names, Internet addresses, and alternate names of hosts on your network that aren't running the *nameserver daemon*. Since all 6130 workstations run the nameserver daemon, those workstations don't need to be listed in this file.

When a network program needs to know the Internet address of another host on your network, the program asks the nameserver daemon on your workstation for that Internet address. The nameserver daemon broadcasts a request for the Internet address of the host over the network. If a nameserver daemon running on another host on the network recognizes the name as its own, the nameserver sends its Internet address back to your workstation.

If no hosts on the network respond to the broadcasted request, the network program looks in the */etc/hosts* file for the Internet address of the host. If that host isn't listed in */etc/hosts*, the network program can't communicate with that host over the network.

Lines in the */etc/hosts* file have the following format:

hostname address [*alias* ...]

where *hostname* is the name by which the remote host is most commonly referenced and *address* is the Internet address of the remote host. The rest of the line contains other optional names you can use to reference the remote host. Any characters after a # (number sign) are considered comments. For more detail on */etc/hosts*, see *hosts(5N)*.

/etc/network.conf

The */etc/network.conf* file contains the network configuration information that is received by the **netconfig** program. You should never have to do anything with this file because it is maintained by the **netconfig** program.

/etc/networks

The */etc/networks* file maps network numbers to network names. The netstat program (discussed later in this section) uses this file to print the name of a network, instead of just its network number, when you are checking the status of another network that you can reach from your network.

Each line in this file has the format:

name number [*aliases*]

where *name* is the official name of the network and *number* is its network number. The rest of the line contains other names by which the network is known. Any characters following a # (number sign) are considered comments.

See *networks(5N)* and *getnetbyname(3N)* in the *UTek Command Reference* manual for more information.

/etc/services

The */etc/services* file contains the names of the network services that are available on your network and the number that the service is referenced by, called its *port*. Various network programs use this file to find out what port to use when requesting a service.

Each line in this file has the format:

service port/protocol [*alias* ...]

where *service* is the official name of the service provided, *port* is the number of the software port that supports this service, and *protocol* is the name of the protocol used. The *aliases* are other names by which the service is known. Any characters after # (the number sign) are considered comments.

See *services(5N)* and *getservbyname(3N)* in the *UTek Command Reference* manual for more information.

/etc/protocols

The */etc/protocols* file maps the names of the network protocols that are used on your network to the number of that protocol. The netstat program uses this file to print protocol names, instead of just protocol numbers, when you request information about that protocol.

Each line in this file has the format:

protocol number [*alias* ...]

where *protocol* is the official name of the protocol provided, and *number* is the number of the protocol. The *aliases* are other names by which the protocol is known. Any characters after a # (number sign) are considered comments.

See *protocols(5N)* and *getprotobyname(3N)* in the *UTek Command Reference* manual for more information.

/etc/tcp_servers

The */etc/tcp_servers* file specifies the network servers that are controlled by the *tcpd network daemon*. Each line in the file has the following format:

name command [*arg* ...]

where *name* is the name of a service that the system performs. See the description of the */etc/services* file in the *services(5N)* manual page for more information on these services. *Command* is either the full pathname or the pathname relative to */etc/tcp_services* of the server that controls the service specified by *name*. *Arg* are arguments that are passed to the server.

Any characters after a # (number sign) are considered comments.

For more on */etc/tcp_servers* and the tcpd daemon, see the Network Daemons topic next in this section and *tcpd(8N)* in the *UTek Command Reference* manual.

NETWORK DAEMONS

A *network daemon* is a program that runs in the background, waiting to handle requests for the network programs. A job that a network daemon handles is called a *service.*

The following paragraphs describe the network daemons that run on your workstation. To make sure these daemons are running, you can use the **daemon** command. For example, the following line reports the status of the *nameserver* daemon:

```
/etc/daemon -v /etc/nameserver
```

See *daemon(8)* in the *UTek Command Reference* manual for more information.

When any of these daemons are handling a request for a service and an error occurs, the daemon calls the **/etc/syslog** program to make an entry in a syslog error file. You should check these syslog error files periodically to make sure the network daemons are working properly. For more information, see *syslog(8N)* in the *UTek Command Reference* manual.

The Nameserver

Your workstation runs a daemon called the *nameserver* that translates names of other hosts on the network to their Internet addresses, provided that host is also running the nameserver.

When you request a service from another host on the net that requires the Internet address of the remote host, the nameserver sends a request out over the network. The nameserver that is running on the remote host reads the request and sends its Internet address back to your workstation.

You can check the status of the *nameserver* with the **/etc/namedbg** command. To use **namedbg**, type:

/etc/namedbg

At the namedbg prompt, which is the asterisk (*), you can enter any of the following namedbg commands:

help	Lists the commands you can use in namedbg.
dump	Lists the names of all the hosts that the nameserver knows about and some information about those hosts.
add	Adds a host to the list of hosts the nameserver knows about.
delete	Deletes a host from the list of hosts the nameserver knows about.
find	Finds an entry in the list of hosts the nameserver knows about.
quit	Exits namedbg.

For more information, see *nameserver(8N)* and *namedbg(8N)* in the *UTek Command Reference* manual.

The tcpd Daemon

The *tcpd daemon* listens for requests for several network services. When a request is received for one of those services, the tcpd daemon starts up the appropriate server to handle the service.

On other operating systems based on Version 4.2 BSD UNIX, one daemon runs in the background for each service that can be requested. When a request is received for one of these services, the daemon monitoring that service handles the request.

For example, on other UNIX-based systems, one daemon runs for the rlogin command, one for the rsh command, one for the FTP service, one for the TELNET service, plus others. These daemons all consume computing resources and therefore slow the system down.

Under UTek, one daemon, tcpd, waits for requests from the rlogin command, the rsh command, the FTP service, the TELNET service, and more. When the tcpd daemon receives a request for one of these services, it starts up the appropriate daemon to handle the request and goes back to waiting for requests. When the requested service is satisfied, the servicing daemon exits.

The services that tcpd handles are specified in the */etc/tcp_servers* file. See the description of that file under the Network Files topic earlier in this section and *tcpd(8N)* in the *UTek Command Reference* manual for more information.

The udpd Daemon

The *udpd daemon* is similar to the tcpd daemon, but handles the simpler network services. These services are time, date, echo, discard, character generator, who is logged in on the system, and which hosts on the network are running. It also handles most NFS services.

Like tcpd, the udpd daemon does the work of several daemons on other UNIX-based systems and therefore takes a load off of the CPU.

See *udpd(8N)* in the *UTek Command Reference* manual for more information.

RUNNING NETSTAT

You can run the netstat program to find out the status of your network, especially if you suspect problems. Netstat has several options that each print different information. The following paragraphs explain the most common netstat options and describe their output.

```
$ netstat -a
Active connections (including servers)
Proto Recv-Q Send-Q  Local Address      Foreign Address    (state)
tcp        0      0  poe.1023           mako.login         ESTABLISHED
tcp        0      0  *.smake            *.*                LISTEN
tcp        0      0  *.echo             *.*                LISTEN
tcp        0      0  *.ftp              *.*                LISTEN
tcp        0      0  *.telnet           *.*                LISTEN
tcp        0      0  *.login            *.*                LISTEN
tcp        0      0  *.exec             *.*                LISTEN
tcp        0      0  *.shell            *.*                LISTEN
udp        0      0  *.tekname          *.*
udp        0      0  localhost.1032     localhost.syslog
udp        0      0  *.talk             *.*
udp        0      0  *.echo             *.*
udp        0      0  localhost.1027     localhost.syslog
udp        0      0  *.who              *.*
udp        0      0  localhost.1026     localhost.syslog
udp        0      0  localhost.1025     localhost.syslog
udp        0      0  *.route            *.*
udp        0      0  *.syslog           *.*
```

Example 3-1. Netstat –a Output.

Calling netstat with the –a argument prints one line for each active network connection from your workstation. See Example 3-2. The columns of the output have the following meanings:

Proto	The protocol used on this connection. These protocols come from the */etc/protocols* file. The two most common protocols are TCP and UDP.
Recv-Q	The number of bytes that are on the Receive queue. This number is usually zero.
Send-Q	The number of bytes that are waiting to be sent out over the network. This number is usually zero. If repeated calls to netstat show this number getting larger and larger, there may be a problem sending data out over the network, although this isn't an accurate gauge of such a problem.
Local Address	A symbolic representation of the address of your workstation. The name is in the form *hostname.socket*, where *hostname* is the name of your workstation and *socket* is the number of the socket or the name of the service that is assigned to that socket. The asterisk (*) is a *wildcard*, meaning there is a network daemon waiting to respond to a request from any host on the network (if the * appears in the hostname part of the address) or on any socket (if the * appears in the socket part of the address).
Foreign Address	A symbolic representation of the address of the other host. This address is in the same format as the local address. If there is no name associated with the address of the foreign host, its Internet address is printed. The asterisk (*) is a *wildcard*, meaning the connection can be made with any foreign hostname (if the * appears in the hostname part of the address) or on any socket (if the * appears in the socket part of the address).
(state)	The state of the TCP connections. ESTABLISHED means that there is communication taking place over the connection. LISTEN means that the workstation is waiting for someone to request a connection on this port. CLOSE_WAIT means that the connection is closing.

```
$ netstat -r
Routing tables
Destination      Gateway          Flags    Refcnt Use        Interface
7.0.0            poe              U        0      25913      lna0
127.0.0          localhost        U        0      77         lo0
teknet           orca             UG       0      0          lna0
8.0.0            orca             UG       0      0          lna0
2.0.0            orca             UG       0      0          lna0
9.0.0            orca             UG       0      0          lna0
3.0.0            orca             UG       0      0          lna0
10.0.0           orca             UG       0      0          lna0
4.0.0            orca             UG       0      0          lna0
5.0.0            mako             UG       0      0          lna0
6.0.0            orca             UG       0      0          lna0
```

Example 3-2. Netstat –r Output.

The netstat –r command shows the route taken by data sent to another network. See Example 3-3. The columns of the output have the following meaning:

Destination — The address of the network. If there is a name associated with this address in the file */etc/networks*, that name is substituted for the address.

Gateway — The name of the host you have to go through to reach the network. The named host may be the gateway to the network, or it may be one of a number of hosts that your data has to go through before reaching the gateway to the network.

Flags — The state of the network. The U flag shows the network is up, and the G flag shows that the named host is the gateway to the network.

Refcnt — The number of active TCP connections on this network.

Use — The number of packets of information that have been sent to other networks over this route since the route was established.

Interface — The name of the interface on your workstation that is used to reach the network.

```
$ netstat -i
Name  Mtu   Network     Address      Ipkts   Ierrs Opkts   Oerrs Collis
lna0  1500  7.0.0       poe          208060  663   26158   11    55
lo0   1536  127.0.0.0   localhost    17457   0     17457   0     0
```

Example 3-3. Netstat –i Output.

The netstat –i command prints one line for each interface on your workstation. See Example 3-4. The columns of the output have the following meaning:

Name — The name of the interface through which data is sent to the network. In Example 3-4, to send data to any other host on the network, the data must be sent through the *lna0* interface. The name *lo0* is the abbreviation for the interface used when a program attempts to send data over the network and the destination of that data is the same workstation (*poe*).

Mtu — Short for *maximum transmission unit*. Mtu is the maximum number of bytes that can be sent over the network in one packet of information. The Ethernet standard says the largest packet of information that can be sent over the network is 1500 bytes (actually 1506, but 6 are used for headers and checksums).

The Mtu for *poe* is larger than 1500 because data sent to this name doesn't go out over the network. For example, if you are on the host named *orca* and you try to rcp (remote copy) some data to *orca* (the host you are already on), the data won't go over the network; it is sent to *loopback*.

Network — The name of the network. This is a symbolic representation of the number of the network, as listed in the file */etc/networks*. If there is no name listed for the network in the */etc/networks* file, the destination is listed in Internet address form.

Address — The hostname of the host through which you access this network.

Ipkts — The number of input packets that have been received from this interface.

Ierrs — The number of input errors that occurred while receiving the input packets.

Opkts — The number of output packets that have been sent to this interface.

Oerrs — The number of errors that occurred while sending the output packets.

Collis — The number of collisions that occurred while communicating with this network. A *collision* occurs when two hosts send data over the network at the same time, resulting in data being lost. When a collision occurs, each host is notified and the transmitter sends the data again.

```
$ netstat -m
194/360 mbufs in use:
     7 mbufs allocated to data
     25 mbufs allocated to packet headers
     55 mbufs allocated to socket structures
     90 mbufs allocated to protocol control blocks
     15 mbufs allocated to routing table entries
     2 mbufs allocated to zombie process information
0/32 mapped pages in use
77 Kbytes allocated to network (32% in use)
0 requests for memory denied
```

Example 3-4. Netstat -m Output.

The netstat -m command shows how much memory the network software is using. See Example 3-5. Each line has the following meaning:

Mbufs in use	An *mbuf* is a small block of memory allocated to the network software. Example 3-5 shows that there are 360 mbufs allocated to the software and 194 of those mbufs are being used.
Mbufs Allocated	Lines 2-7 of the output show how those 194 mbufs are being used. Data is information being passed by users over the network. Packet headers, socket structures, protocol control blocks, and routing tables are data structures used by the network software. Zombie processes are processes that are exiting but are not totally finished yet.
Mapped Pages	A *mapped page* is a large block of data allocated to the network software. In Example 3-5, there are 32 such blocks allocated to the network software and none of them are being used.
Kbytes Allocated	The total amount of memory (in kilobytes) allocated to the network software.
Requests Denied	If the network software requests more memory and the system can't provide that memory, then some network data is lost. This is not necessarily a problem, since in many cases the data is retransmitted, so the data loss is only temporary and the data is recoverable.

NETWORK ADMINISTRATION TASKS

When workstations are connected together with a LAN, there are concerns, such as proper internet addressing, that become network-wide. To be sure these concerns are properly dealt with, you should get together with the system administrators of the other workstations on the LAN and make one person in charge of these details. This person is called a *network administrator*.

The responsibilities of the network administrator are:

- Determining a class of Internet addressing to be used on the network. See the Ethernet and Internet Addresses topic earlier in this section for more information.
- Making sure hostnames and Internet addresses are unique for each workstation on the network. As a network administrator, you can simplify the tasks of the system administrators on your network by assigning hostnames and Internet addresses to their workstations.
- Assigning ranges of userids and groupids for each system administrator to use so that network security is preserved. Also, the network administrator expands ranges of userids and groupids when necessary. See the Assigning User and Group Identifications topic earlier in this section for more information.
- Making sure that all workstations on the network are synchronized with respect to time.

In addition, you may want to set one workstation up on your network as a *file server node*. See the discussion on File Server Administration for more information.

FILE SERVER ADMINISTRATION

If your workstation has a 61TC01 streaming cartridge tape drive (with or without the optional hard disk drive that can be ordered with the 61TC01), you may want to set your workstation up as a *file server node.*

Most often, a fileserver node is a host on the network to which other hosts are backed up. A file server node can also be dedicated to specific tasks that require large amounts of memory or peripherals. For example, a file server node might store a data base, such as an inventory record or bill of material lists for engineering projects, on its mass storage devices that other workstations on the network access through the distributed file system. If the network is small, the same file server workstation might have the hard copy devices, such as a printer and plotter, for the entire network connected to it. Hard copy requests from other workstations on the network are then sent to the file server, using MDQS, for output. (Instructions for configuring the MDQS to send jobs over a network are in Section 4).

Figure 3-10 shows a representation of a file server node in a network.

Using the File Server for Back Ups and Restores

CAUTION

When you are using the **wsdump** *and* **wsrestore** *commands to back up and restore data over the network, all systems participating in the back up or restore must be in multi-user mode. Unlike backing up to local media, multi-user mode is required for network utilities to be enabled. You can prevent back up problems that result in loss of data by making sure the workstations being backed up or restored are in a quiescent state. That is, only the system administrator is logged in the workstation initiating the back up or restore, and all other workstations participating in the back up or restore are absent of unnecessary user logins, user processes, or jobs. To make periodic back ups over the network easier, you should take back ups during periods of low activity for your network, such as late at night or early in the morning.*

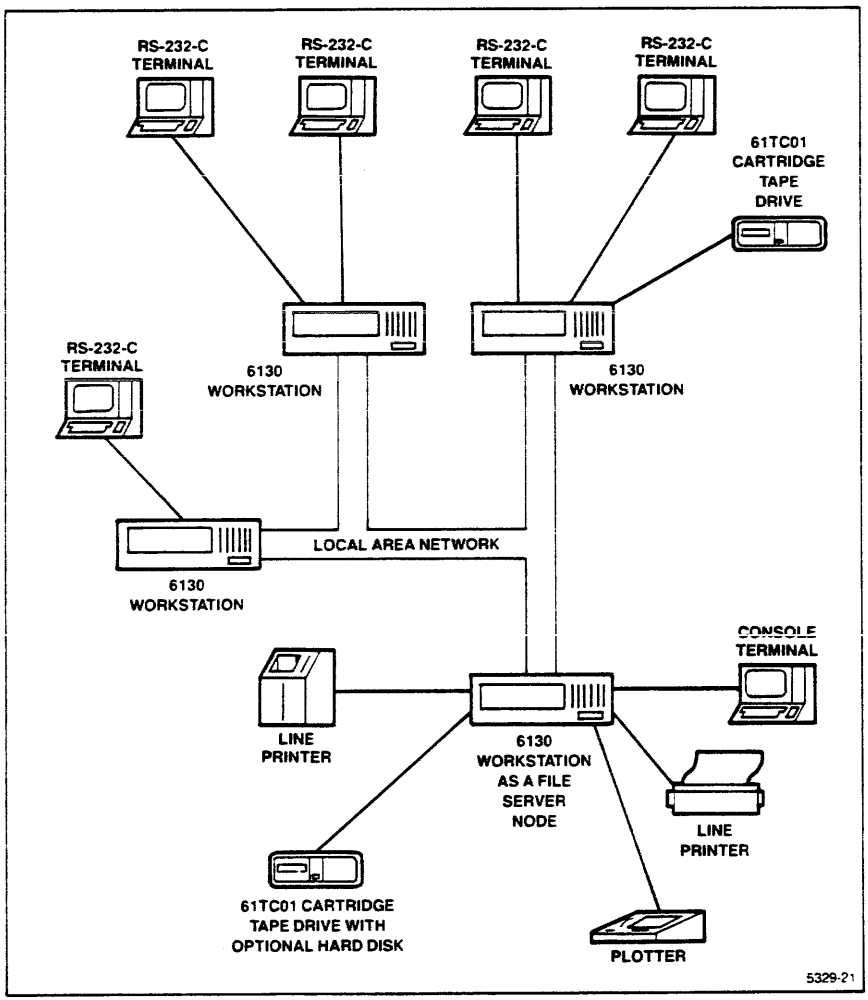


Figure 3-8. File Server Node

If the file server node is used to back up other workstations on the network, the stored files must be written to an external peripheral such as the 61TC01 streaming cartridge tape drive. A file server node can be configured to automatically initiate the back up process of other workstations on the network. This means that no one on the workstations being served have to be involved with those workstations while they are being backed up. However, an operator should be present at the file server workstation to change media when necessary.

NOTE

If you have workstations on the network with software earlier than Version 2.2 (or later) UTek, you should copy the Version 2.2 /usr/bin/rdump and /usr/bin/rrestore programs to those workstations. The wsdump program uses the version 2.2 rdump and rrestore programs, and earlier versions of these programs may not be compatible with the later versions.

Backup software consists of:

- the backup utilities *dump* and *restore* and the remote backup utilities *rdump* and *rrestore*
- the programs *wsdump*, *wsrestore*, and *wsass* to control *rdump* and *rrestore* from the file server node
- the */etc/wsdumptable*, */etc/wsdumpdates*, and */usr/adm/wsdumplast* system files
- the *.netrc* file in the *dumpopr* account
- the program *viwsb*, which is used to edit */etc/wsdumptable*

Wsdump runs on the file server and remotely executes a back up on the workstations served. If all tables and files are properly set up, *wsdump* handles the back up process, including choosing what workstations to back up at what levels, logging into served workstations, backing up served workstations in sequence, and tape space and volume allocation. As with any other back up procedure, the file server administrator is notified when to change media. *Wsdump* should only be executed on the file server from the *dumpopr* account.

NOTE

The dumpopr account is set up at the factory on all new workstations shipped with Version 2.2 (or later) UTek. If you are upgrading your workstation to Version 2.2 UTek, you must set up the dumpopr and dumprmt accounts when you do the upgrade. Appendix C 64WP02 UTek Reinstallation tells you how to do this.

Wsdump reads */etc/wsdumptable* (and the *.netrc* file) to automatically perform back up scheduling and login functions. *Wsdumptable* defines the frequency of the different possible dump levels and */usr/adm/wsdumplast* defines what group of workstations have been most recently dumped. *Wsdumptable* also defines the dump requirements of each workstation or file system. See *wsdump(8n)* and *wsdumptable(8n)* in the *UTek Command Reference* manual for more information. *Wsdumptable* is also discussed in more detail later in this section.

The *.netrc* file is a list of machine names and passwords for the *dumpopr* and *dumprmt* accounts of those machines. Once the *wsdump* program has determined what workstations to back up (based on the */etc/wsdumptable* file), it looks in here to find out the passwords of the *dumpopr* accounts on the remote workstations. When the remote workstations are ready to execute the back up, they find out the password of the file server's *dumprmt* account from their local *.netrc* file. The *.netrc* file is discussed in more detail later in this section.

The file */etc/wsdumpdates* is a record of all dumps made. The file */usr/adm/wdumplast* is a record, by group, of the last dumps made. *Wsdump* uses these files in conjunction with *wsdumptable* to determine which workstations need to be backed up.

Wsrestore remotely executes *rrestore* on the workstations requiring file restorations. *Wsrestore* searches *wsdumpdates* for appropriate volume and dump level information. The *wsass* program automatically assigns the device that you specify using *wsrestore*.

Setting Up the File Server

One of the first things you must do when setting up the workstation as a file server is to make sure that all other workstations periodically synchronize their clocks with the file server clock (see the discussion on the timed daemon earlier in this section for more information). Having workstation clocks synchronized to the file server is critical to the make utility and other functions that rely on a specific time. Also, as you will see in the Running wsrestore Example (later in this section), all workstations should maintain the same time so that the times recorded for back ups are the same for the file server workstation and the workstations being served.

To execute *wsdump*, the *dumpopr* account must have special files on both the file server workstation, and the workstations being served. Workstations must have these files and directories before the file server administrator can use the **wsdump** and **wsrestore** commands:

- the */usr/dumpopr/.netrc* file — on the file server and the workstations being served
- the */usr/dumpopr/restore* directory — on workstations being served
- the */etc/wsdumptable* file — on the file server workstation

Setting up the .netrc Files

NOTE

You can either set up the .netrc file on the served workstations yourself, or you can instruct the system administrators of those workstations to create the .netrc file.

To set up the *.netrc* files:

1. Log into each workstation, including the file server, as *dumpopr*.
2. Create a file in */usr/dumpopr* called *.netrc*. The contents of the *.netrc* file is one line of information. This line of information contains the file server name and the password of the *dumprmt* account on that workstation. Each line in the *.netrc* file should be in this form:

   `machine` *name*`, login dumprmt, password` *dumprmt_password*

   where *name* is the name of the file server workstation, and *dumprmt_password* is the password of the *dumprmt* account for the file server workstation.

3. On the fileserver workstation only, the *.netrc* file should have additional entries for each workstation being served in the form:

   `machine` *name*`, login dumpopr, password` *dumpopr_password*

   where *name* is the name of a workstation being served, and *dumpopr_password* is the password of the *dumpopr* account for that workstation.

For example, assume there are four workstations on a network: *huey*, *dewey*, *louie*, and *donald*. *Donald* is the file server for *huey*, *dewey*, and *louie*.

Every other night, the administrator of *donald* backs up all four workstations (including *donald*) by logging into the *dumpopr* account on *donald* and running *wsdump*. When the *wsdump* command line is entered, the administrator specifies a 61TC01 cartridge tape connected to *donald* as the media device.

Every workstation on the network has a */usr/dumpopr* account, with the passwords set to **quack1** (donald), **quack2** (huey), **quack3** (dewey), and **quack4** (louie). Each workstation on the network also has a */usr/dumprmt* account. However, the only *dumprmt* account that is important is the one on the file server (donald), which has a password of **ducks**.

The *.netrc* file in the *dumpopr* account on *donald* contains names of all the workstations it serves, and looks like this:

```
machine donald, login dumprmt, password ducks
machine donald, login dumpopr, password quack1
machine huey, login dumpopr, password quack2
machine dewey, login dumpopr, password quack3
machine louie, login dumpopr, password quack4
```

The *.netrc* files on *huey*, *dewey*, and *louie*, which are served by *donald*, look like this:

```
machine donald, login dumprmt, password ducks
```

Setting Up the /etc/wsdumptable File On the file server workstation, you must set up a file called */etc/wsdumptable* (a skeleton */etc/wsdumptable* already exists, but you must fill in the proper information for your network). The *wsdumptable* file is read by *wsdump* to find out the schedule for back ups of the served workstations.

To create the *wsdumptable* file:

1. Log in to the file server workstation as *root*.
2. Type **viwsb**. This command displays the *wsdumptable* file and lets you fill in information using the **vi** editor. Example 3-6 shows an empty *wsdumptable* file you get when you first use **viwsb**.

Wsdumptable consists of two parts separated by a dotted line. The top part of the file contains information on how often, in days, to back up at each level. The bottom part of the file contains information on each filesystem to be backed up. A completed *wsdumptable* may appear similar to Example 3-7.

Groups are defined in the bottom half of the file, and are denoted by a single letter. A group is a subset of all the workstations to be served. The purpose of setting up groups is to prevent *wsdump* from trying to back up all file systems on all workstations in the same back up session. *Wsdump* does not distinguish groups with upper and lower case letters, so that workstations belonging to group *a* are the same as those belonging to group *A*. If you have only a small number of workstations on your network, you may wish to put them all in the same group.

Note that the use of the use of the word *groups* in conjunction with the *wsdumptable* file is different than the concept of *user groups* discussed in Section 4.

In this example, the top half of the file tells *wsdump* that:

- Level 0 back ups are done every 30 days. Level 0 back ups are done to a maximum of one group a day.
- Level 1 back ups are done every seven days. Level 1 back ups are done to a maximum of two groups a day.
- Level 9 back ups are done every day. Level 9 back ups are done to a maximum of nine groups a day.

For more information on levels of backup, see the discussion in Section 4.

```
#
# $Header: 03,v 1.2 87/06/15 14:34:21 archive Exp $
#
# Workstation Backup : Dumptable
#
#    Header Definition area
#
#     dump level : frequency : max groups to dump
#
------------------------------
#
#    Workstation Entry Definition area
#
#     workstation name : file system : dump levels : group
#
```

Example 3-5. Empty Wsdumptable.

In Example 3-7 the bottom half of the file tells *wsdump* that:

- The filesystem */dev/dw00a* on *huey* participates in back ups on levels 0, 1, and 9, and is a member of group A.
- The filesystem */dev/dw00a* on *dewey* participates in back ups on levels 0, 1, and 9, and is a member of group B.
- The filesystem */dev/dw00a* on *louie* participates in back ups on levels 0, 1, and 9, and is a member of group B.

For more information on *wsdumptable*, see *wsdump(8n)* and *wsdumptable(8n)* in the *UTek Command Reference* manual.

As an example of how the *.netrc* and *wsdumptable* files work, assume the file server administrator logs into *donald* as *dumpopr* and types **wsdump /dev/ntc64**. When **wsdump** executes, the *wsdumptable* file on *donald* is read. This file tells *wsdump* the frequency of back ups for the workstations on the network. Using the *wsdumptable* and *wsdumplast* files, *wsdump* determines which workstations to back up. *Wsdump* looks at the *.netrc* file on *donald* to find the passwords for the *dumpopr* accounts on these workstations.

Wsdump automatically logs into the individual *dumpopr* accounts on the workstations being served and runs *rdump* on those workstations. When *wsdump* calls a workstation, it tells the workstation that the remote back up should use the *dumprmt* account of *donald*, and that the remote device should be */dev/ntc* (a cartridge tape drive). When a workstation is ready to do a back up, it looks in its *.netrc* file for the password of the *dumprmt* account on *donald*. Because the device has been specified as a cartridge tape drive on *donald*, this is the media on which the information is backed up (see the later example on running *wsdump*). When the media is full, and a new volume must be inserted in the drive, *wsdump* informs the file server operator.

```
#
# $Header: 03,v 1.2 87/06/15 14:34:21 archive Exp $
#
# Workstation Backup : Dumptable
#
#    Header Definition area
#
#     dump level : frequency : max groups to dump
#
0:30:1
1:7:2
9:1:9
-----------------------------
#
#    Workstation Entry Definition area
#
#     workstation name : file system : dump levels : group
#
huey:/dev/dw00a:019:A
dewey:/dev/dw00a:019:B
louie:/dev/dw00a:019:B
```

Example 3-6. Completed Wsdumptable.

Setting Up the *restore* Directory In the *dumpopr* account on the workstations being served (*huey*, *dewey*, and *louie* in the previous example), you should set up a directory called *restore*.

The purpose of the *restore* directory is to give the file server administrator a place to put a restored file on one of the workstations being served, without having to have *root* permission on the remote workstation. The system administrator of the remote workstation can then log in as *root* and restore the file to the proper user account.

To set up the *restore* directory, do this procedure on each workstation being served:

1. Log in to the workstation as *dumpopr*.
2. Type:

```
mkdir restore
```

Continuing with the earlier example, assume that a user on *louie* accidently removed a file. The user tells the system administrator about it, in hopes of getting the last backed-up version of the file restored. The administrator of *louie* then contacts the administrator of the file server *donald*. The fileserver administrator of *donald* uses the wsrestore command to find and restore the file on *louie* in the */usr/dumpopr/restore* directory. The system administrator of *louie* then logs in as *root*, and moves the restored file from */usr/dumpopr/restore* to the directory of the user who lost the file.

To see how the *restore* directory is used, see Running wsrestore Example.

Running wsdump Example

Once you have set up the files and directories necessary for a file server, you can use the **wsdump** command to periodically back up workstations on the network.

To execute **wsdump**:

1. Make sure that the media you are using for back up is in the proper drive. For example, if you are using the 61TC01 tape drive as a back up device, make sure you have a tape in the drive.
2. Log into the file server workstation as *dumpopr*.

   NOTE

   The command **wsdump** *has many options not shown in this example. For more information on* **wsdump**, *see* wsdump(8n) *in the* UTek Command Reference *manual.*

3. Type:

   ```
   /etc/wsdump /dev/ntc64
   ```

   This command line backs up all filesystems on all workstations specified in the */etc/wsdumptable* file to the cartridge tape device */dev/ntc64*.

Running wsrestore Example

Once you have set up the files and directories necessary for a file server, you can use the **wsrestore** command to restore files from the file server to other workstations on the network. *Wsrestore* is an interactive program that asks you questions about the files you want to restore.

To execute *wsrestore*:

1. Log into the file server workstation as *dumpopr*.

   NOTE

   The command **wsrestore** *has many options not shown in this example. For more information on* **wsrestore**, *see* wsrestore(8n) *in the* UTek Command Reference *manual.*

Type:

```
/etc/wsrestore -S
```

The workstation then displays:

```
Enter restore device name (/dev/default):
```

2. In response to this prompt, enter the device you want to restore from. For example, if you want to restore from the cartridge tape drive connected to your workstation, you would answer:

   ```
   /dev/ntc
   ```

   in response to this question (*/dev/ntc* specifies a cartridge tape with no rewind). If you wish to select the default device, press <RETURN>.

   The default device is determined by how you enter the command line. The command line shown in this example, which uses the -S option, defaults to a streaming cartridge tape (*/dev/tc*) as the device. The -F option to **wsrstore** (typing **/etc/wsrestore -F**) defaults to the flexible disk drive. If you type **/etc/wsrestore** with no options, the default is a nine track tape drive.

   Once you have entered the device, this prompt displays:

   ```
   Enter workstation name:
   ```

3. Enter the name of the workstation you want to restore to. This prompt displays:

   ```
   Enter filesystem to restore:
   ```

4. Enter the name of the filesystem you want to restore on the workstation you specified. For example, you might type:

   ```
   /dev/dwØØa
   ```

   in response to this question. This prompt displays:

   ```
   Enter path of directory to restore to (usr/dumpopr/restore):
   ```

5. To restore to the */usr/dumpopr/restore* directory on the remote workstation (the default), press <RETURN>. You can also specify another directory in response to this prompt. Once you have specified the directory to restore to, this prompt displays:

```
Enter restore options (i):
```

6. Enter <RETURN> to accept the **i** (interactive restore) option to the **restore** command. The interactive restore option lets you look at the restore media and select specific files to restore using a shell-like environment. At this prompt, you also have the choice of specifying other options to **restore**. For more information on options to **restore**, see *restore(8)* in the *UTek Command Reference* manual. The interactive restore mode is discussed in more detail in Section 4 of this manual, and *restore(8)* in the *UTek Command Reference* manual.

   Once you have specified the **restore** option(s), this prompt displays:

```
Enter date last modified:
```

7. Enter the date the file you want to restore was last modified. You can respond to this question in many ways. For example, these are all acceptable answers:

   - **2/6/86**
   - **February 6, 1986**
   - **February 6 5:30 PM**
   - **yesterday**
   - **today**
   - **one hour ago**
   - **10:08 PM**

   For more information on other ways to specify the time, see *getdate(5mdqs)* in the *UTek Command Reference* manual.

   Once you have entered the date the file was last modified, this prompt displays:

```
Enter date file lost:
```

8. Respond to this prompt in the same form that you specified time in the previous question. When you have entered the date the file was lost, *wsrestore* searches */etc/wsdumpdates* to see if there is a back up of the filesystem you specified for the time you specified.

   a. If **wsrestore** finds a record of a back up on that date, this information displays:

      ```
      Load volume #n on dump device device_name
      Ready [y/n]?
      ```

      where *n* is a volume number, and *device_name* is the name of the device that made the back up.

      Locate this volume and place it in the drive. When you have done this, press <RETURN>, and continue to the next step.

   b. If **wsrestore** can find no record of the dump on the date you specified, this prompt appears:

      ```
      Try another date [y/n]?
      ```

   c. If you answer **y**, **wsrestore** asks you the previous two questions again. This way, if the first date you gave provides no record of a back up, you can search for back ups made on different dates until to find a back up of the filesystem you are looking for.

      If you do not want to search other back ups, answer **n** to this question. **Wsrestore** quits this back up, and displasy the **Restore another workstation/filesystem [y/n]?** prompt (see the discussion in the last step).

9. When you have found the volume of the back up and placed it in the drive and pressed <RETURN>, the workstation begins the restore of the filesystem you specified, to the workstation(s) you specified. If you specified the interactive option to **restore**, you are placed in interactive mode. This allows you to pick and choose the files from that filesystem that you want to restore.

10. When the restore is complete, this prompt displays:

    ```
    Restore another workstation/filesystem [y/n]?
    ```

    If you answer **y** to this question, **wsrestore** takes you back to the *Enter workstation name:* prompt. **Wsrestore** then goes through the prompts again for the second workstation or filesystem. In this way, you can use **wsrestore** to restore multiple filesystems in the same session.

    If you answer **n** to this question, **wsrestore** returns the system prompt.

Restoring from a Served Workstation

If a user on a workstation being served by the file server loses a file, it is not necessary for the file server administrator to use *wsrestore* to restore the file. Rather, the administrator of the workstation that lost the file can use the **rrestore** command to restore the file themselves, if the dump number and volume number of the back up is known.

To use **rrestore** in this manner:

1. Log into the workstation that lost the file as **dumpopr**
2. Change directories to the *restore* directory. Type:

   ```
   cd /usr/dumpopr/restore
   ```

3. Execute the **rrestore** command. An example command line for **rrestore** is:

   ```
   /etc/rrestore ifsb donald.dumprmt:/dev/ntc 2 63
   ```

   in this command line:

   - i specifies interactive mode
   - f specifies to log in to the workstation *donald* using the *dumprmt* account, and to use the cartridge tape device */dev/ntc* (specified by the line **donald.dumprmt:/dev/ntc**).
   - s specifies back up number 2
   - b specifies a blocking factor of 63

There are many more options to the **rrestore** command than what is shown in this example. See *rrestore(8n)* in the *UTek Command Reference* manual for more information.

Sysadmin Interface Procedures

INTRODUCTION

This section discusses system operations you can perform using the *sysadmin interface*. Many of the common system operations are made easier with this interface. The sysadmin interface is a combination menu and form-fill-out system that allows you to perform system administration tasks without having to know UTek thoroughly. The interface reads and write files for you, so you do not have to manually edit the files yourself. Operations you can do using the sysadmin interface include:

- Configuring queues for devices.
- Setting network parameters.
- Configuring RS-232-C ports.
- Writing a message of the day.
- Setting up the table of system daemons.
- Configuring the intermachine mail system.
- Configuring the UUCP system.
- Performing system backups and restores.
- Adding applications and optional software.
- Adding, maintaining, and deleting users.
- Adding, maintaining, and deleting groups.

This section assumes that you understand some things about UTek:

- The basic file structure of UTek and how to move about in it.
- The concepts of file protections and file ownership.
- How to use one of the system text editors (preferably vi).

You can find information on these subjects in *Introducing the UNIX System* by Henry McGilton and Rachel Morgan, in the *6130 Learning Guide*, and in the online sessions.

Also, the concepts underlying many of the procedures discussed in this section are covered in Section 6 of this manual and in volumes one and two of the *UTek Tools* book.

Figure 4-1 shows the tasks covered by the sysadmin interface, and the organization of the tasks within the menus.

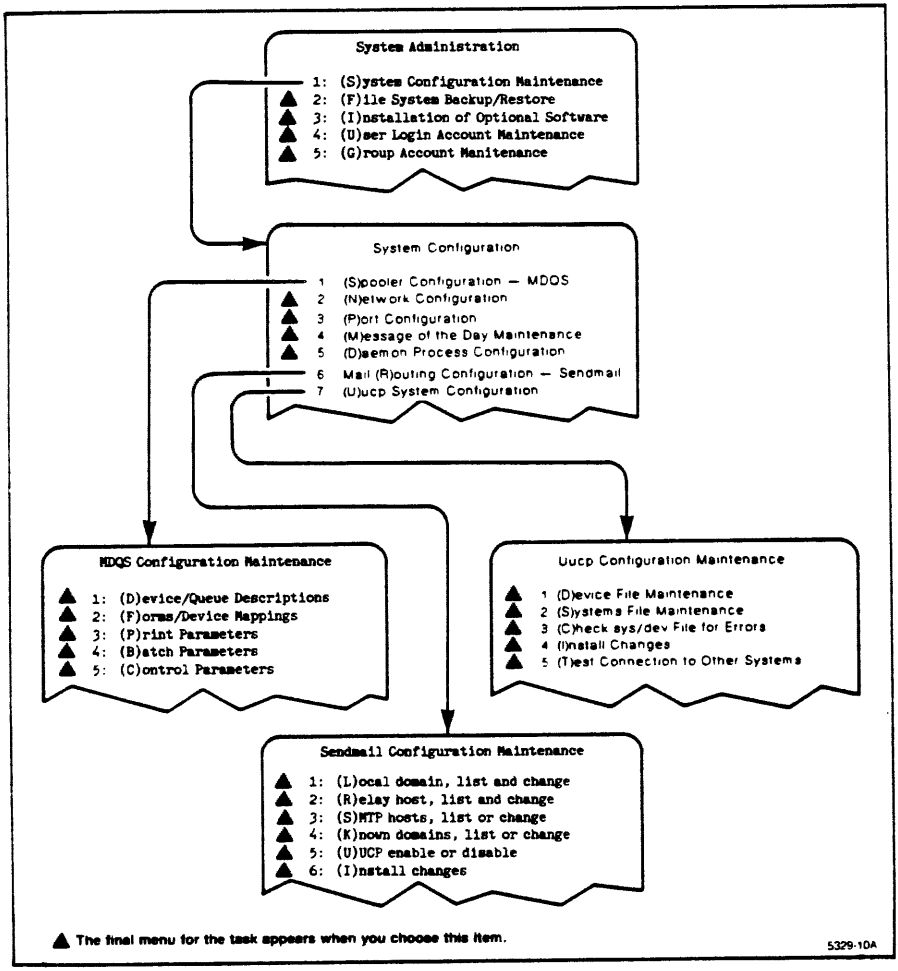


Figure 4-1. Organization of the Sysadmin Interface.

USING THE INTERFACE

Open this manual to the discussion of the menu(s) for a task the first time you perform that task. Reading the manual as you work teaches you the sysadmin interface and provides you with information about how you should respond to sysadmin prompts.

Getting Into the Interface

There are three ways to get into the sysadmin interface:

- Log into the *sysadmin* account.
- Log in as *root* and type:

  /etc/sysadmin

- Type:

  /etc/sysadmin

  from your regular account, and enter the sysadmin password when asked.

Before logging you in, the sysadmin interface asks you for your terminal type. After responding to this prompt with the proper terminal type, the top-level sysadmin menu is displayed.

It doesn't really matter which of these methods you use. It mostly depends on whether you are logged on already when you decide to use the sysadmin interface, and what account you are logged in to.

Moving Around in the Interface

The interface is made up of a two-part screen. At the top are menus listing various tasks, and at the bottom is a workspace, which usually contains a form to be filled in.

To access a task, type the number of the task, or the letter in parentheses (either upper or lower case) associated with the task.

You can also type **q** almost any time to get out of **sysadmin**. The interface asks you if you really want to quit, in case the **q** was an error and you don't want to wait for the interface to set up again, or you haven't saved the changes you've made.

If you are in the workspace at the bottom, you can usually press the <ESC> key to get to the menu at the top. When you're at the menu level, pressing the <ESC> key takes you back to the previous menu, or out of the interface, if you are at the top level.

If you want to use a UTek command while you are inside the interface (for example, **more** to view the contents of a file), preface the command with an exclamation mark (!). When the UTek command is finished, you are returned to the interface.

Since the system does not redraw the screen after the UTek command is finished, you may want to redraw the screen yourself by typing <CTRL-L> while in the menu space.

Asking for Help

Sysadmin has a help facility, which you can access by typing **?**, **H** or **h**. You can get help on the interface itself, help on a task, help on the options for a task, or help on the help facility. When you ask for help, the prompt line changes. Enter the number or letter of the topic on which you want help, or enter **?**, **H** or **h** again for general help.

Don't be afraid to use sysadmin help to find out more. Often, the help screen contains more specific information about the topic than what is given in this manual. Use both sysadmin help and the discussions in this manual to give you a better understanding of the topic.

To get out of help, you must press <ESC>. This redisplays the menu and allows you to make active menu choices.

THE MENUS

The organization of this discussion reflects the organization of the interface (refer back to Figure 4-1). The menu structure branches from the top-level menu down to the individual tasks. This discussion follows these "branches" as shown in Figure 4-1, beginning with *Spooler Configuration*, under *System Configuration Maintenance*, moving on to *Network Configuration*, and so on.

Special Considerations

Some of the procedures performed by this interface require special preparations before you use them. These are:

- Backup and Restore: the system should be in single user mode before you take a backup or restore files.
- Uucp Configuration: Special files must be set up (details in the Uucp Configuration discussion).

SYSTEM CONFIGURATION MAINTENANCE

This menu selection covers topics relating to setting up the system for regular operation. The subjects include:

- Queues for devices that require them, using the Multidevice Queuing System (MDQS)
- Network parameters.
- RS-232-C port configurations for the devices that you are connecting to them.
- Message of the Day as a means of communicating with users.
- System daemon configuration.
- Mail routing configuration.

Spooler Configuration — MDQS

MDQS stands for Multidevice Queuing System. It provides a means of sending printing or batch requests to devices that you define (defining devices is discussed in Section 5). MDQS arranges the order of tasks and sends them to a variety of devices. The most common use for MDQS is to send printing jobs to a printer. The *batch* queue takes files that contain UTek commands and sends them one at a time to the shell for processing.

The file that contains the information that MDQS uses is */etc/qconf*. The menu options in the MDQS Configuration Maintenance menus modify this file. Example 4-1 shows the original 6130 */etc/qconf* file.

A detailed discussion of the MDQS is available in the *UTek Tools* book.

```
#       MDQS configuration file
#
#       Parameters
#
debug           4
console         /usr/spool/q/qtmp/mdqs.log
openwait        10
scanwait        60
netwait             10
maxfailures     10
sysmgr          root
mdqsid          mdqs
hostname        ariel
#
#  Batch queue parameters
#
batch-forms     Shell
batch-queue     batch
batch-prior     5
#
#  Print queue parameters
#
print-forms     narrow
print-queue     lp
print-prior     5
print-hdr       /usr/lib/mdqs/lphdr
print-limit   0
#
#  Versatec queue parameters
#
# NOTE: These items can not be viewed or changed by the
# System Administration interface.
#
----------------------
#
#  Device Descriptions
#
# <dname>     <device>      <forms>              <status>
batch0          /dev/null  Shell
----------------------
#
#  Queue Descriptions
#
# <qname>
batch
----------------------
#
```

```
#  Queue->Device Mappings
#
# <qname>     <dname>              <server>
batch           batch0          /usr/lib/mdqs/shserver
```

Example 4-1. Sample /etc/qconf File.

Some Definitions

The *mdqs daemon* is a process that signals *server processes* that a job is waiting in a *queue* to be processed. The mdqs daemon is started when other system daemons are started at system boot. See the discussion on daemons in Section 6.

A *queue* is a data structure in the MDQS that sequences requests that are then processed in an order based on both the order in which they were received and their priority.

A *physical device* is the MDQS terminology for the device file in */dev*.

A *logical device* is used by MDQS software to reference devices in the */dev* directory. Each logical device must be defined in the */etc/qconf* file to correspond to a device file in */dev*.

The UTek commands that send jobs to MDQS queues are the lpr command and the batch command. Descriptions of these commands can be found in *lpr(1)* and *batch(1)* in the *UTek Command Reference* manual. Lpr sends jobs to the queue described in the Print queue parameters section of the */etc/qconf* file. Batch sends jobs to the queue described by the Batch queue parameters in the same file. You can set both these sets of parameters using these MDQS menus.

Server programs, also known as *server processes*, run each time a job leaves a queue. These are the programs that direct the job to the proper device. When you set up a queue-to-device mapping (that is, associate a queue with a logical device), you must assign a server program to the mapping. There are a limited number of server programs. These reside in the directory */usr/lib/mdqs*. When you add a queue-to-device mapping using the sysadmin interface, you must specify one of these server programs to serve the jobs that leave that queue.

Device/Queue Descriptions

This menu selection lets you add or delete device and queue descriptions in the */etc/qconf* file. It lets you map queues to devices, and assign server programs to the mappings. You can also list the currently defined queues. If you connect a printer to the workstation, you should add a queue for that printer.

There are two ways to add a queue: set up a queue-to-device mapping for a device on your workstation, or set up a mapping to a line printer on another host on your local area network (a remote host).

List Queue Descriptions The first thing you should do when you enter the Device/Queue Descriptions menu for the first time is list the queue descriptions. This gives you an idea of what the interface expects and how the */etc/qconf* file associates queues with logical devices, logical devices with files in the */dev* directory (physical devices), and all of these with the server program that serves them.

If you are preparing to delete a queue description, you should list the descriptions first and note the number of the description you want to delete. Then use the Delete option to delete that number description.

Add Queue Descriptions Use this selection when you want to add a new queue to the workstation. The fields you have to fill out are:

Queue Name — Choose a name that is descriptive of the device to which the queue maps. For example, queues for line printers should end with *lp*.

A queue name can be from 1 to 14 alphanumeric characters, and should start with an alphabetic character.

Logical Name — The name of the logical device to which the queue you just specified maps. A logical name can be from 1 to 14 alphanumeric characters, and should start with an alphabetic character. Logical devices are explained earlier in this section.

Device Name — The name of the device file in the */dev* directory (also known for MDQS purposes as the physical device) with which the logical device defined above associates. For batch and network logical devices, this physical device should be */dev/null*. For other logical devices, the physical device file must already exist before you assign a logical device to it. Setting up physical devices is discussed under *Adding Devices* earlier in this section.

Server Program — Assigns the program that serves the queue and device when there is a print request. Table 4-1 indicates which server program you should choose depending on the type of device to which the queue is sending data. Three of the servers (*lpserver*, *plpserver*, and *rawlpserver*) are for specific printers or types of printers. For more information on these servers, see the *lpserver(8mdqs)* manual page.

To leave this option without adding a queue, or to leave after you are finished adding a queue, press <ESC>.

Table 4-1
AVAILABLE SERVER PROGRAMS

Server Program	Device
/usr/lib/mdqs/shserver	batch
/usr/lib/mdqs/netsend	network
/usr/lib/mdqs/lpserver	character printers
/usr/lib/mdqs/plpserver	Printronix printers
/usr/lib/mdqs/rawlpserver	graphics printers
/usr/lib/mdqs/tcpserver	GRF images

Delete Queue Description This selection lets you delete a queue-to-device mapping from the system. List the queue descriptions first, note the number of the mapping you want to delete, then choose this Delete option and enter that number in the *Delete number* field and press <RETURN>.

If you want to delete the queue-to-device mapping that is used by the lpr command, you must choose the Print Parameters option on the *MDQS Configuration Maintenance* menu first, and erase all the fields in the workspace (change them to empty by pressing <RETURN> instead of filling in an entry). Otherwise, you get an error when you try to delete the mapping.

To leave this option without deleting a queue, press <ESC>.

Add Remote Lineprinter Queue Description This selection lets you add a queue description that maps to a line printer on another host on your local area network. Enter the hostname of the host you wish to access into the Add machine field, and the interface fills in the queue name, logical device, physical device, and server program for the queue description.

Since this interface is only adding information to the */etc/qconf* file, it does not check if the hostname you enter is a valid one, so make sure you are using a valid hostname for your network. Hostnames are discussed in more detail under the Network Configuration heading later in this section.

The queue name assigned to the remote lineprinter queue is the hostname of the remote host (as you entered it) followed by lp. For example, a remote lineprinter queue to the remote host *engr1* would be *engr1lp*. The logical device is set to *net* and the physical device is set to */dev/null*. The server program for all remote queues is */usr/lib/mdqs/netsend*.

When you specify a request to the remote queue with the lpr or batch commands, the forms default to the standard forms as set by the Print Parameters or Batch Parameters menu, respectively.

If the forms assigned to the remote printer by the remote host are not the same as the default forms in your workstation's print parameters, jobs sent to the remote queue do not get processed on the remote host.

To correct this, find out what forms are assigned to the remote printer by the remote host (ask the system administrator of the remote host, or look in the remote host's */etc/qconf* file). If they are not the same as forms in your workstation's print parameters, specify the correct forms for the remote printer with the -f flag of the **lpr** or **batch** commands when you type in the request.

If the date setting on your workstation is later than the date setting on the remote host, jobs sent to the remote host may take longer to process than originally expected. For example, if the date on your workstation is July 15, and the date on the remote workstation is July 1, jobs sent from your workstation to the remote host are marked for processing on July 15, and therefore wait in the remote host's queue until its date setting reaches July 15. In other words, the job takes two weeks to process.

The date and time settings on your workstation and on the remote hosts should be as close as possible, so that they always have the same calendar day.

Use the **date** command to find the date on your workstation and on the remote host. If they do not match, use the **date** command to reset the dates appropriately. For information on **date**, see *date(1)*.

To use the remote line printer, your workstation must either be in the */etc/hosts.equiv* file on the remote host, or users must have login accounts with the same login name and userid on the remote host, with the proper entries in their *.rhosts* files. See Section 3 for discussions of the */etc/hosts.equiv* and *.rhosts* files.

To leave this option without adding a remote queue description, press <ESC>.

Forms/Device Mappings

This menu option lets you view and set the forms for a logical device. All logical devices except network devices must have forms assigned. Devices can only have one type of forms, unlike queues, which use forms to indicate to which logical device they should send a request.

Valid forms assignments can be from 1 to 8 alphanumeric characters, and should begin with an alphabetic character. Example forms are *wide* and *narrow*. These could be used to specify the paper width on two different printers that get requests from the same queue.

List Device/Forms Mappings Lists the current forms for all devices currently in queue descriptions. Use this option to get the number of the Device/Forms mapping you want to change.

Change Device/Forms Mapping To change the forms for a device, use the List option to get the number of the mapping you want to change, enter that number when asked, then fill in the name of the forms you choose.

Print Parameters

This menu selection controls the default behavior of a printing job: which queue is the default print queue, what printing forms are used, the default priority of print requests within the queue, the file that is used as the header for print jobs, and the maximum number of pages that can be printed in one print job.

Queue Name	The name of the default print queue. This queue receives print jobs from the lpr command when no -q argument is given. You must assign this queue to the printer device. You can do this using the *Device/Queue Descriptions* selection of the MDQS Configuration Maintenance menu.
Forms	Sets the default forms for this queue. These are the forms that a print request is assigned if the lpr command that generated the print request doesn't have a -f option. Forms are a way that MDQS distinguishes between different devices if a queue holds jobs for more than one device. See *Forms/Device Mappings* in this section for details on forms.
Priority	Sets the default priority for print requests. The priority of a print request determines its position in the queue. This number should be from 0-10. The highest priority is 0, and the lowest priority is 10. A good default priority is 5.
Banner File	Specifies the file that contains additional information to be printed on banner pages. The banner is printed at the beginning of each print request along with the standard header. If you don't put anything here, this field defaults to */usr/lib/mdqs/lphdr*, which is an empty file.
Banner Directory	The directory where MDQS looks for a file specified by a -H *filename* option for an lpr command. See *lpr(1)* in the *UTek Command Reference* manual for more information on the -H option.
Page Limit	The maximum number of pages to be printed per print request. Any requests for more pages are denied. This prevents people from printing long jobs on active printers. The page limit number can be any number from 0 (no limit) to 32767.

Batch Parameters

This menu selection sets the values for parameters in the batch queue. For more information, see *batch(1)* in the *UTek Command Reference* manual.

The parameters you can set with this selection are the default queue and the priority.

Queue Name — The name of the default batch queue. A batch request goes to this queue if a user doesn't specify a different queue with the -q option of the **batch** command. You must assign this queue to the batch device. You can do this using the *Device/Queue Descriptions* selection of the MDQS Configuration Maintenance menu.

Priority — Sets the default priority for batch requests. The priority of a request determines its position in the queue. This number should be from 0-10. The highest priority is 0, and the lowest priority is 10. A good default priority is 5.

Control Parameters

Choosing this menu lets you set the parameters that control the actions of the MDQS daemon. The form that appears in the workspace shows you the current values of the parameters, and each selection on the menu lets you change a parameter.

Console File Name — Redirect the standard error output to this file.

Openwait (seconds) — If the MDQS daemon cannot open a device (the device may be busy), the daemon waits this many seconds before trying to open the device again.

Scanwait (seconds) — Sets the amount of time the MDQS daemon is idle after a check of all devices that it serves. On an idle system, this affects how often the daemon checks delayed requests.

Failure Threshold — Sets the maximum number of times a server process can fail before the device it serves is marked as "failed." If the device fails, you can restart the device using **qdev**. (See *qdev(8).*)

System Manager Name	Sets the login name of the MDQS system manager. The daemon mails messages about MDQS operation to this address. The default login name is *mdqs*. Make sure the MDQS system manager has a login account.
Netwait	If a network request fails (such as a print request to a remote printer), the spooler waits the amount of time specified by the *netwait* parameter before retrying. The value given by this parameter is in minutes. The number should be between 0 and 32767 (10 is the default).

Example of MDQS Configuration

Here is an example of how to set up a queue using the menus just discussed.

Assume that you want to connect a matrix printer to your workstation, and that you want to connect it to the standard RS-232-C port 2. Remember that UTek addresses this port as */dev/tty11*. Also assume that you want this printer to be the default printer. Use the following procedure to set up a queue so that when you use the lpr command without arguments, the specified file is printed on the printer you are installing here.

You should be logged in as *root* or *sysadmin* and have the *MDQS Configuration Maintenance* menu on your screen. The procedure for setting up the print queue uses this menu.

1. Create a Device/Queue Description. This step sets up the print queue and associates the queue with a physical device (the port) and the device's server.

   To create this description:

   a. Choose menu item 1, Device/Queue Descriptions.

   b. When the *MDQS Queue Maintenance* menu appears, choose item 2, Add Queue Description.

c. The following form appears in the workspace:

```
Adding new queue description.

     Queue Name      :
     Logical Name    :
     Device Name     :
     Server Program  :

     Enter <ESC> to abort addition
```

Assign an entry for each field in the form. You have to devise a Queue Name and a Logical Name, but the Device Name and the Server Program are defined by the port to which you are connecting the printer and the type of printer, respectively.

d. Pick a queue name that follows the rules previously stated (1 to 14 alphanumeric characters, start with alphabetic character). Since you are connecting a printer here, call the queue **lp** (for "lineprinter").

e. Pick a logical name that follows the rules previously stated (1 to 14 alphanumeric characters, start with alphabetic character). Since this printer is going to be the default printer, call the logical device **lp0**. The logical name is the name MDQS associates with the physical device Device Name. All references within MDQS to the physical device are made using the Logical Name.

f. Because you are connecting this printer to RS-232-C port 2, the device name here must be **/dev/tty11**. This field should always be a device file in the */dev* directory.

g. Because you are connecting a matrix printer, the server program here must be **/usr/lib/mdqs/plpserver**. This field must be one of the server programs in Table 4-1.

h. After you are back at the menu level of the *MDQS Queue Maintenance* menu, press <ESC> to get back to the *MDQS Configuration Maintenance* menu.

2. Next, set the Forms/Device Mapping for the printer. This is the way you assign forms to the logical device you just created. Forms are defined earlier in this section.

   Choose item 2 on the *MDQS Configuration Maintenance* menu. The *MDQS Device/Forms Mappings* menu appears on the screen.

   a. Choose item 1 on the *MDQS Device/Forms Mappings* menu to list the current mappings. You do this because you need the to find out the number on this list of the logical device you just created in order to set the forms for the device.

   b. Remember the number for the logical device **lp0** in the Logical Device column of the list of mappings.

   c. Choose item 2 on the *MDQS Device/Forms Mappings* menu.

   d. When the **Change number:** prompt appears, put in the number for the **lp0** logical device and press <RETURN>.

   e. A new prompt line appears listing the device and prompting you to set the forms. The forms you set for a printer should represent the type of paper in the printer. In this example, assume you have 8 1/2 by 11 inch paper, and set the forms to *narrow*. Type in **narrow** and press <RETURN>.

   f. Press <ESC> to return to the *MDQS Configuration Maintenance* menu.

3. This step sets the default queue, and forms for the lpr command. When you invoke the lpr command in the future, the print job is sent to the printer specified by the parameters you set in this step. Choose item 3 on the *MDQS Configuration Maintenance* menu, Print Parameters.

   The *MDQS Print Parameter Maintenance* menu appears in the menu space and the following form appears in the workspace:

```
Queue Name       :
Forms            :
Priority         :
Banner File      :
Banner Directory:
```

   a. Choose item 1 of the *MDQS Print Parameter Maintenance* menu. Type in ip (the queue you defined in step 1) for the queue name and press <RETURN>.

   b. Choose item 2 of the *MDQS Print Parameter Maintenance* menu. Type in narrow (the forms you defined in step 2) for the forms and press <RETURN>.

   c. Choose item 3 of the *MDQS Print Parameter Maintenance* menu. Type in 5 for the default priority of jobs you send to the printer with the lpr command.

   d. For this example, do not choose either item 4 or item 5. Item 4 already has a default file associated with it, and you only need item 5 if you plan to use the -H argument for the lpr command (see *lpr(1)*).

   e. Press <ESC> to return to the *MDQS Configuration Maintenance* menu.

4. Type S or s to save the changes you just made to the MDQS configuration file and return to the *System Configuration* menu.

At this point, the printer you attached to standard RS-232-C port 2 has a print queue associated with it, and is the default printer for the lpr command.

Network Configuration

Choosing this selection lets you set or change the hostname and internet address of the workstation, turn off or on the Network File System (NFS) and standard network capabilities, and change the address(es) of the workstation's LAN interface.

If you change any of these, restart the network daemons. The network daemons are:

- syslog
- nameserver
- routed
- udpd
- tcpd
- talkd
- inetd
- portmap

Restart these daemons using the **daemon** command (see *daemon(8)*), or by rebooting the system. If you don't restart the daemons, network operations on your system won't work correctly. Details on restarting daemons are discussed in Section 6.

This menu item basically performs the function of the **netconfig** utility. Detailed information on **netconfig** and on configuring and using a LAN is available in Section 3, Lan Administration.

Change Host Name

Select this menu item to change your workstation's hostname.

The hostname is the name by which other users on the network can address your workstation. It is associated with an internet address (discussed later).

The hostname must be an alphanumeric string 1 to 32 characters long. The hyphen (-) and underscore (_) characters are also valid. The first character of a hostname must be a letter. Details on choosing a hostname are discussed in Section 3.

If you don't want to change your workstation's hostname, press <ESC> or <RETURN> instead of entering a new hostname.

See *hostname(1N)* for more information on the hostname.

Change Host id

This menu item lets you change the workstation's *hostid*. The hostid is a number that uniquely identifies your workstation among workstations on a Network File System. This number is the internet address of LAN interface *ilan0* by default, and the recommended procedure is that you let it remain at this default. However, you can change the hostid to a number of the form:

w.x.y.z

Where *w*, *x*, *y*, and *z* are any number from 0 to 255 (inclusive).

If you don't want to change the hostid, press <ESC> or <RETURN> instead of entering a new hostid.

See *hostid(IN)* and the hostid discussion in Section 3 for more information.

Toggle Network Utilities

This menu selection changes the state of the standard network utilities. If the utilities are on, choosing this item turns them off. If the utilities are off, choosing this item turns it on.

Changing the state of the standard network utilities with this menu item does not kill the network daemons if you toggle the standard network utilities from on to off, nor does it start network daemons if you toggle from off to on. To kill the daemons use the **kill** or **daemon** command, to start it, use the **daemon** command or reboot the system. It is not necessary to kill the daemons, but it may free some system resources. The daemons are not started the next time the system is started.

Change Interface Addresses

Select this menu item to change the internet address for the workstation's LAN interface. When you choose this item, another menu appears.

List Internet Addresses This item lists all the LAN interfaces on the workstation and their corresponding internet addresses. There is always at least one entry for *lna0*.

The list is numbered, and you can use the list entry number to specify an interface during the Change Internet Address part of this menu (discussed next).

Change Internet Address This item lets you change the internet address of a LAN interface on your workstation.

When you choose this item, you are prompted with `Change entry:`. Type either the list entry number of the interface whose internet address you want to change, or the interface's name.

After you indicate the interface whose internet address you want to change, a line appears in the workspace prompting you to enter the new internet address for the interface. Type in the new address and press <RETURN>.

The internet address is the address by which other hosts on the LAN address your workstation. It is associated with your workstation's hostname.

The internet address must be of the form:

w.x.y.z

where *w*, *x*, *y*, and *z* are all decimal numbers between 0 and 255 (inclusive). See Section 3 for a discussion on classes of internet addresses and on choosing an internet address.

If you don't want to change a LAN interface's internet address, press <ESC> or <RETURN> any time you are prompted for input, and you return to the menu.

Port Configuration

This menu choice deals with the RS-232-C ports and the pseudoterminal ports used for network logins. Ports are usually associated with terminals and other devices (such as lineprinters), and have file counterparts, which are usually stored in the */dev* directory. Using this selection, you can see what parameters are assigned to which ports, list the type of ports, list the terminal types, delete a port, or assign or change parameters and configurations of ports.

When you assign a configuration to a port, you are basically assigning a baud rate to the port. Make sure that the device you connect to the port is set to the same baud rate as the port, or you might encounter communication difficulties.

When you select the *Port Configuration* item from the *System Configuration* menu, the files */etc/gettytab* and */etc/termcap* are read to obtain information about terminal and port types. These files are only read by the interface, and cannot be changed through this menu. The files */etc/ttys* and */etc/ttytypes* are also read when you enter this interface, but you can change the contents of these files through this menu.

List Port Information

This menu selection lists information for each port on the system. For each port, this information is listed:

Port name	This is the name the port is known by in the */dev* directory. It must be unique and have 255 or fewer alphanumeric characters. Once a port name exists, you can delete it or change it with the *Delete Port* or *Change or Add Port* menu items.
Terminal type	This is the abbreviation for the terminal type as specified in the */etc/termcap* file. For example, *4105* means the Tektronix 4105 terminal. You can find out what the terminal types are by selecting *Terminal Type List* from the Port Configuration Menu.

Login (yes or no)	If a user can log in from this port, this field should contain a yes. If the port has a line printer or some other nonlogin device connected to it, this field should contain a no.
Configuration Type	This one-letter code defines the type of configuration (baud rate) assigned to the port. To find out what the codes mean, select *Port Type List* from the Port Configuration menu.

Each port information entry has an entry number, which you can use when you delete or change a port (see the *Delete Port* and *Change or Add Port* discussions).

Port Type List

This menu selection provides a list of the port types. For example, this list tells you that port type *y* is a standard 9600 baud port.

Terminal Type List

This menu selection provides a list of terminal types as defined in the */etc/termcap* file. For example, this list tells you that *tek4014* is a Tektronix 4014 graphics terminal. This terminal types list is very long. If you do not want to view the entire list, you can get back to the menu by pressing <ESC>.

Delete Port

This menu selection asks you *Enter port to be deleted.* You can then respond with the port you want to delete. Enter the entry number or port name and press <RETURN>. The interface then deletes the configuration information for a port from the files */etc/ttys* and */etc/ttytype*.

You can obtain a list of ports by using the *List Port Information* selection from the Port Configuration menu.

If there is an entry for the for the port *console* in the */dev* directory, you cannot delete it using this interface.

Change or Add Port

If you change terminals, attach a line printer, or change the communications parameters of the connected terminal, you should change the configuration of the port concerned.

This selection lets you change the configuration of an already existing port or add a new port and its corresponding configuration menu information (however, you cannot modify the port *console* using this interface). Configuration information is in the files that store port information, */etc/ttys* and */etc/ttytype*. After you create the device file in */dev* for a port (see Section 5), you should add the port to the configuration files using this interface.

When you select this menu item, the interface prompts you with

```
Enter port to be changed:
```

If you want to change port information, enter the entry number or port name of an existing port. If you want to add information, enter the name of a not yet configured port that has and entry in */dev* (you can find out what ports exist by using the *List Port Information* selection from the Port Configuration menu). Once you have selected a port number or port name, a menu is displayed in the menu space. the The four existing port parameters (if you are changing a port) or the four change categories without the parameters (if you adding a port) are displayed in the workspace.

The *Configuration Change or Add* menu contains four categories:

Change (N)ames	— use this item to change the port name. The new name must not be *console* or the name of an existing port. The new name must also be less than 255 characters and contain no spaces or tabs.
Toggle (L)ogin Status	— use this item to toggle the login status from *yes* to *no*, or from *no* to *yes*.

Change (P)ort Type — use this item to change the port type. The new port type can be the port type character (as listed by the *Port Type List* item in the parent menu), or the form *@port*, where *port* is the name of an existing port. If you use *@port*, the port type of that port is used for the new port type.

Change (T)erminal Type — use this item to change the terminal type for the port. The new terminal type can be given as a terminal type (as listed by the *Terminal Type List* item in the parent menu), or the form *@port*, where *port* is the name of an existing port. If you use *@port*, the terminal type for that port is used for the new terminal type.

Using these menu items, you can selectively change or add one or all of the port parameters.

Saving Changes

To leave the *Port Configuration* menu without saving changes, press <ESC>. To leave the menu and save the changes you have made in */etc/ttys* and */etc/ttytype*, type **S** or **s**.

After you make changes to the files for new RS-232-C ports (if you installed the Dual RS-232-C interface enhancement board), reboot the workstation to ensure proper operation of the new ports.

Message of the Day Maintenance

By using the Message of the Day (motd), you can get a message to all users of the workstation. The message appears on each user's screen at login. You can use the motd to announce scheduled downtime for system backup or maintenance, or ask people to delete unnecessary files if storage space is running low. This message is stored in the */etc/motd* file, which you can create, change or delete whenever you want. The system checks when a user logs in to see if the file is there and, if it is, prints the contents.

With the sysadmin interface, you can list, delete, edit or create a new message of the day. If you create a new motd, it overwrites the current contents of the file. If you just want to add a line or two to the current message, choose the Edit option.

In the Create mode, you can enter text directly, in which case you can enter only 60 characters per line. In this mode, indicate blank lines by typing a space on the line and pressing <RETURN>.

In the Edit mode, you can enter the editor of your choice. Edit the motd as you would any other text file, and exit the editor as you would normally. You are *not* out of the sysadmin interface at this point — when you exit the editor, the menu reappears.

Daemon Process Configuration

This menu selection deals with setting up the */etc/daemontab* file. It lets you view the contents of this file, edit the file, and check the command lines in the file to make sure they are syntactically correct.

The */etc/daemontab* file contains commands that define what is to be done with system daemon processes when you run the **daemon** command.

List Contents of Daemon Description File

This menu selection prints the */etc/daemontab* file in the workspace. You might want to do this in order to see what daemons are installed on the system, or to get information on what arguments can be used if you want to add a daemon.

Edit Daemon Description File

This menu selection lets you edit the */etc/daemontab* file. You might want to edit this file in order to add a command line to the file, or to *comment out* a command line.

Commenting out means putting a # (number sign) at the beginning of a line so that it is treated as a comment instead of a command. For example, if you aren't connecting your workstation to a network, there's no reason for the network daemons to run. Commenting out the command lines for the network daemons frees the computing resources that the daemons would otherwise use by running constantly in the background.

When you choose this item, the interface puts you into **vi** unless you have another default editor set with the EDIT or EDITOR environment variables in your login environment file. See the *6130 System User's Guide* for details on setting these environment variables. With **vi**, the menu disappears and you have full screen editing. You are *not* out of **sysadmin** at this point — when you exit the editor, the menu reappears.

Check and Explain Daemon Description File

This menu selection prints the */etc/daemontab* file, and gives a line by line explanation of it.

The explanation tells you how the line is interpreted by the **daemon** command, and also tells you if there are any syntax errors in command (non-comment) lines.

Mail Routing Configuration — Sendmail

Sendmail is a program that lets your workstation communicate with other workstations. This menu lets you set up and change the contents of configuration file for **sendmail**, located in */usr/lib/sendmail.cf*. For a discussion on the background and concepts involved in using **sendmail**, see Section 6.

Local Domain, List and Change

The local domain defines the hosts with which users on your workstation can exchange mail. Users can only exchange mail with users on all hosts within the local domain and other known domains.

This menu option tells you the current local domain, as listed in the *sendmail.cf* file, and lets you change it if you want to. If you don't want to enter a new local domain, press <RETURN> or <ESC> to return to the menu. If you want to enter a local domain, type in the domain you want and press <RETURN>.

The domain name should be 1 to 32 alphanumeric characters long and begin with a letter. The hyphen (-) and underscore (_) characters are also valid.

To find out the local domain you are in, ask the system administrator of a host already in the domain. If you just created a network and are setting up mail between hosts on a new network, all system administrators on the network should choose a name for the local domain.

Relay Host, List and Change

This menu option tells you the current *relay host*, and allows you to change relay hosts. A relay host is the host that messages addressed to unknown hosts are sent, in hopes that the relay host knows about the destination host. The relay host must be in one of the lists of known hosts (SMTP or UUCP).

The name of the relay host must be a valid hostname. That is, the name must be from 1 to 32 alphanumeric characters (beginning with a letter), with the hyphen (-) and underscore (_) characters also being valid.

SMTP Hosts, List and Change

This menu option causes a new menu to appear that lets you:

- List the hosts reachable over the LAN currently in the configuration file.
- Add a host to the list.
- Delete a host from the list.
- Change a hostname in the list.
- Erase the entire list of hosts.

List All Smtp Hosts This option lists the hosts currently reachable over the LAN.

Add a Smtp Host This option lets you add a hostname to the list of Smtp hosts. Type in the name of the host you want to add, then press <RETURN>. When you are finished entering hosts, press <RETURN> or <ESC> to get back to the menu.

The hosts you add to the list must be valid hostnames. That is, the name must be from 1 to 32 alphanumeric characters (with the name beginning with a letter); the hyphen (-) and underscore (_) characters are also valid.

Delete a Smtp Host This option lets you delete a hostname from the list of Smtp hosts. You can only delete one host at a time. If you decide you don't want to delete a host from the list, press <RETURN> or <ESC> to get back to the menu.

Change a Smtp Host This option lets you change the name of a host in the list. If a host in the list changes name for some reason, use this option to change the entry.

The hostnames you add to the list must be valid hostnames. That is, the name must be from 1 to 32 alphanumeric characters (with the name beginning with a letter); the hyphen (-) and underscore (_) characters are also valid.

Erase All Smtp Hosts This option lets you remove all hosts from the list of Smtp hosts. When you choose this item, the interface asks you if you really want to erase all the hosts. Answer y (for yes) if you want to, and n (for no) if you don't want to erase all Smtp hosts.

Known Domains, List and Change

This menu option causes a new menu to appear that lets you:

- List the known domains in the configuration file.
- Add a domain to the list.
- Delete a domain from the list.
- Change the name of a domain in the list.
- Erase all the known domains from the list.

List All Known Domains Lists the domains currently in the configuration file. The domain LOCAL should always be in this list. If it isn't, you should add it, as the LOCAL domain is necessary for proper local (on the workstation) mail delivery.

Add a Known Domain This option lets you add a domain to the list of known domains. Type in the name of each domain you want to add, then press <RETURN>. When you are finished entering domains, press <RETURN> or <ESC> to get back to the menu.

The domains names you add must be from 1 to 32 alphanumeric characters (with the name beginning with a letter); the hyphen (-) and underscore (_) characters are also valid.

Delete a Known Domain This option lets you delete a domain from the list of known domains. You can only delete one domain at a time. If you decide you don't want to delete a domain from the list, press <RETURN> or <ESC> to get back to the menu.

Change a Known Domain This option lets you change the name of a domain in the list. If a domain in the list changes name for some reason, use this option to change the entry.

The domain names you add to the list must be from 1 to 32 alphanumeric characters (with the name beginning with a letter); the hyphen (-) and underscore (_) characters are also valid.

Erase All Known Domains This option lets you remove all domains from the list of known domains. When you choose this item, the interface asks you if you really want to erase all the domains. Answer y (for yes) if you want to, and n (for no) if you don't want to erase all known domains.

UUCP Enable or Disable

NOTE

The system information that allows you to use uucp is in the /usr/lib/uucp *directory. This directory is part of the UTek/A optional software package. If UTek/A is not installed, you cannot communicate with other hosts using uucp. For more information on uucp, see the* Uucp System Configuration *discussion later in this section, and the uucp discussion in the* UTek Tools *book.*

This option tells you whether mail using uucp is enabled or disabled, and lets you enable or disable mail over uucp. If you don't have the proper uucp software on your workstation, uucp should be disabled. The default setting for uucp mail is "disabled."

To enable uucp, answer y to the question *Do you wish to enable UUCP mail?*. To disable uucp, answer n to this question. To get back to the menu without doing anything, press <RETURN> or <ESC>.

Install Changes

This menu option updates the */usr/lib/sendmail.cf* file to reflect any changes you made with the previous menu options. You should always select this option before you leave the *Sendmail Configuration Maintenance* menu if you made any changes to the configuration file.

If you don't choose this menu option, and you made changes to the configuration file, a message appears in the workspace when you try to leave the *Sendmail Configuration Maintenance* menu telling you that you have made changes and asking you if you want to update the configuration file. If you don't want to lose the changes you made, enter y in response to this question.

Example of Mail Routing Configuration

Here is an example of how to configure mail for your workstation using the menus just discussed.

Suppose you want to set up the sendmail configuration file */usr/lib/sendmail.cf* so that users on your workstation could exchange mail with users on other hosts (the *sendmail.cf* file already allows mail between users on the same workstation). There are two ways that a 6130 workstation can send and receive outside mail: Smtp and uucp. Smtp mail requires the workstation to be connected to a LAN (see Section 3), and uucp mail requires that the 64WP05 UTEK/A package be installed, and that uucp itself be configured and operational.

This example assumes that your workstation is connected to a LAN, and that uucp is *not* installed. Follow these steps to configure the workstation for outside mail.

You should be logged in as *root* or *sysadmin* and have the *Sendmail Configuration Maintenance* menu on your screen.

1. Choose item 1 on the *Sendmail Configuration Maintenance* menu to assign a local domain. A local domain is necessary if you want to communicate with other hosts. This form appears in the workstation:

```
The local domain is <None Specified>
Type:
     <CR>  - to leave unchanged.
     <ESC> - to abort this command.
     New local domain name -->
```

   Suppose you are connected to a new network, and all the system administrators on the network have agreed to name the local domain **GENERAL**.

   In the form that appeared in the workspace type:

   GENERAL

   and press <RETURN>.

2. Choose item 2 on the *Sendmail Configuration Maintenance* menu to assign a relay host. If you assign a relay host, anytime a user sends mail to an unknown host, it goes to the relay host to be forwarded. This form appears in the workspace:

```
The relay host is <None Specified>
Type:
     <CR>  - to leave unchanged.
     <ESC> - to abort this command.
     <DEL> to have no relay host.
     New relay host name -->
```

   Suppose the host *engr1* on your network is connected to another network as well. Use engr1 as the relay host by typing into the form:

   engr1

   and pressing <RETURN>.

3. Choose item 3 on the *Sendmail Configuration Maintenance* menu. This causes the *Sendmail List Maintenance* to appear.

   At this point you want to create the list of known hosts, so choose item 2 on the *Sendmail List Maintenance* menu. The following form appears in the workspace:

```
Enter new Smtp hosts
Type:
     <CR> or <ESC> to finish entering.
     Smtp host to add -->
```

   Type in the names of all the workstations in the local domain. For example:

   engr1
   engr2
   engr3a
   eval1
   eval2b

   Type <RETURN> after each entry, then <RETURN> or <ESC> when you are finished. Be sure and put the relay host (engr1, here) into the list of hosts.

4. Type <ESC> to get back to the *Sendmail Configuration Maintenance* menu.

5. Make sure that uucp is disabled by choosing item 5 of the *Sendmail Configuration Maintenance* menu. If the form in the workstation says:

```
Uucp mail is disabled
Type:
     <CR>  - to leave unchanged.
     <ESC> - to abort this command.
     Do you wish to enable UUCP mail?
```

   press <RETURN>. If the form says:

```
Uucp mail is enabled
Type:
     <CR>  - to leave unchanged.
     <ESC> - to abort this command.
     Do you wish to enable UUCP mail?
```

   type n, and uucp mail is disabled.

6. Choose item 6 on the *Sendmail Configuration Maintenance* menu. This installs the changes you just made in the *usr/lib/sendmail.cf* file.

Your workstation is now configured to exchange mail with all the hosts in the list you entered in step 3.

Uucp System Configuration

NOTE

This selection is only valid if you have the 64WP05 UTek/A optional software package installed on your workstation. If you try to use these menus without the UTek/A optional software package installed, errors occur.

Using this menu selection, you can change the contents of the files */usr/lib/uucp/L.sys* and */usr/lib/uucp/L-devices*. The contents of these files control the transfer of files and mail to other systems that also have the uucp utility. The *L.sys* file specifies the systems your workstation can communicate with using uucp. The *L-devices* file contains descriptions of all direct connected lines and modems that uucp can use. Each device type and baudrate for each system in the *L.sys* file should have a corresponding entry in the *L-devices* file.

Device File Maintenance

Selecting this menu item lets you list, add, or change the devices contained in */usr/lib/uucp/L-devices*.

The contents of the *L-devices* file contains information about devices uucp can use to place calls to other systems. When uucp wants to call another system, it searches this file to find out how to connect to the given system. The devices and information contained in this file consist of:

- device type
- tty line
- baud rate
- modem brand
- program name

  Select the *Device file maintenance* menu item, then use the help to find out more about filling in or interpreting the fields in this file.

Systems File Maintenance

Selecting this menu item lets you list, add, or change the list of systems contained in the */usr/lib/uucp/L-sys* file.

Each entry in the *L.sys* file represents one system that can be called by the uucp program on your workstation. More than one line may be present for a particular system. For example, there may be additional lines that represent alternative communication paths that will be tried in sequential order if the previous communication path fails. Each line, which represents a particular system, contains these fields:

- remote system name
- permissible call times
- device type
- baudrate
- phone number
- login stream

Select the *Systems file maintenance* menu item, then use the help to find out more about filling in or interpreting the fields in this file.

Check sys/dev File for Errors

Selecting this menu item cross-checks the contents of the *L.sys* file with the contents of the *L-devices* file. The device type and baudrate listed with each system in the *L.sys* file should have a corresponding entry in the *L-devices* file.

If this check finds an error, you should change the system entry in *L.sys* or add a device of that type and baudrate to the *L-devices* file.

Install Changes

After you have modified the uucp configuration data, install the changes using this menu selection. If you try and exit the menu after you have made changes to *L.sys* or *L-devices*, but not installed them with this menu selection, you are asked if you want to install the changes.

Test Connection to Other System

Selecting this menu item lets you check the connection to a remote system. A call is attempted to the remote system. The interface informs you of the result.

FILE SYSTEM BACKUP/RESTORE

CAUTION

The system should be in single-user mode when you are backing up or restoring data to or from local media. If the system is in multiuser mode when you back up data, an imperfect copy of the data may be written to the backup media, resulting in problems when you try and restore from that media. If the system is in multiuser mode when you try to restore data, one or more files may not be properly restored.

If you are backing up or restoring data over the network, your system must be in multi-user mode for the network utilities to be enabled. However, you can minimize any problems you may have with backup and restores by ensuring that the system administrator is the only one logged on, and no user jobs are in process (that is, the system is in a quiescent state).

CAUTION

If something should happen to your system, such as hard disk crash, service personnel can restore the UTek operating system, but cannot restore any personal files if they have not been backed up. Therefore, you should back up your system on a regular basis.

One of the most important tasks you perform as a system administrator is periodic backups of the system. You do this so that if anything happens to the system you can restore a system that is reasonably close to the one you lost. You should back your entire system up on a regular basis such as once a week or even once a day (if the workstation is used heavily).

Also, if there are multiple users on your workstation, users will probably come to you asking for individual files to be restored. The Restore options offered by this menu lets you restore individual files or the entire system from backup media or over a network.

When you back the system up, what you're doing is copying files off the Winchester disk into another file, onto diskettes, or, if you have the SCSI enhancement and a streaming cartridge tape drive on your workstation (or a remote host), onto streaming cartridge tape. There are two menu choices for backup. You can either back the system up to local media on your workstation, or you can back the system up to media on a remote host connected to a network.

When you restore the system, you are copying files from diskette or a streaming cartridge tape to the Winchester disk on your workstation. Again, the source media can be local or on a remote host connected to a network.

The sysadmin interface uses the **dump** command to back up files locally or the **rdump** command to back files up using a remote host. The sysadmin interface uses the **restore** command to restore them from local media or the **rrestore** command to restore from media connected to a remote host. See *dump(8)*, *rdump(8n)*, *rrestore(8n)*, and *restore(8)* for details on these commands.

List Filesystems That Have Been Dumped

This menu item provides you with a list of files that have been *dumped* (that is, backed up) at least once by executing **/etc/dump** with the **W** option. The information displayed by this menu selection is taken from the */etc/dumpdates* and */etc/fstab* files. For more information, see the *dump(8n)* manual page in the *UTek Command Reference* manual.

Backup to Local Media

This menu item runs the **dump** command with the f and u options. The f option sents the dumped information to the device you specify, and the u option is necessary for incremental dumps. For more information on these options, see the *dump*(8) manual page in the *UTek Command Reference* manual.

The sysadmin interface asks you what device you want to use to back up your data. The only choices displayed are for devices that are installed on your workstation. When the sysadmin interface is entered for the first time, the peripherals list is checked to see which peripherals are available. The first device to be found is set up as the default medium. After that, the last device you used becomes the default medium. The default choice is displayed in the workspace above the other device choices. To do a backup to the default choice, press <RETURN>. Possible devices choices are:

- file
- diskette
- 9-track tape
- streaming tape cartridge (QIC-24 format)

The 6130 workstation has a diskette as a standard device. You can choose the streaming tape cartridge, which is displayed if you have the 61TC01 streaming cartridge tape drive installed.

File

If you choose to back up to a file, the interface asks you for the level number, the name of the filesystem you want to back up, and the name of the file to back up to. When the interface asks you for a response, a default value is also displayed where applicable. You can select the default value by pressing <RETURN>, or select another response from those displayed.

When the interface asks you for the level of back up you would like to do, enter a number between 0 and 9. The backup system on the workstation has ten levels of backup: level 0, which is a total system backup, and levels 1 through 9, which record all files that were changed since the previous backup with level number less than or equal to the specified level number. For example, if you take a backup of level 5, it records all files that were changed since the last level 5 backup.

NOTE

The root filesystem, /, is mounted on the device /dev/dw00a. / is the only filesystem standard on the 6130.

When the interface asks you for the name of the file to back up to, the filesystem table is searched for the name of the filesystem in which the file resides. This way, you don't have to know the name of the filesystem, just the the name of a file. The filesystem or filename must be that of an existing file, or you receive an error.

If the file you want to back up to is in the filesystem you want to back up, you also receive an error. For example, if you specify */usr/joeu/back* as the file to backup to, then specify that you want to back up the filesystem /, you receive an error because */usr/joeu/back* is in / (/ is both the default and the only filesystem standard on the 6130).

Once you have entered all the parameters, the interface displays the information you have entered, giving you one last chance to abort the backup if the information is not correct. You can abort the backup by pressing <ESC>. If the information is correct, press <RETURN> to execute the backup.

The interface then takes the answers you have given to the prompts and uses them as arguments of the **dump** command. **Dump** sends you messages on the date of this backup according to the workstation's internal clock, and when the last backup of this level occurred.

Diskette or Cartridge

If you choose diskette or streaming tape cartridge, the interface asks you the same questions as if you had backed up to another file (see the previous discussion). The interface asks you for the level of backup and the name of the filesystem you want to backup.

If you are backing up the system to diskettes, **dump** tells you approximately how many formatted diskettes (**dump** calls them *volumes*) you need to perform the requested backup. Be sure you have this many formatted diskettes available. If you do not, halt the backup by answering **no** to the first **NEEDS ATTENTION** prompt, and then **yes** to the resulting **Do you want to abort?** prompt and get some more formatted diskettes. Information on how to format diskettes is available in Section 5.

If you are using diskettes, **dump** prompts you to change diskettes. When the new diskette is in the drive and the drive door is closed, type **yes** in response to the **NEEDS ATTENTION** prompt. Label the diskettes in order as they come out of the diskette drive, so that if you need to restore from them, you know the proper order.

Restore from Local Media

Sometimes users lose files by one means or another. They then come to you, the system administrator, and ask you to restore the lost file(s). You can save them a lot of work if you can restore a day-old version rather than a week-old version of a file that they work on often.

This *Restore from Local Media* menu item puts you into an interactive mode that lets you restore one, some, or all files in a file system, but is most useful for restoring one or some files. See the *restore(8)* manual page in the *UTek Command Reference* manual for more details on restoring the system.

The first thing the interface asks you when you choose *Restore from Local Media* is what device you want to restore from. Like the *Backup to Local Media* menu selection, on the devices present on your system are displayed. Possible choices of devices are:

- file
- diskette
- 9-track magnetic tape
- streaming tape cartridge (QIC-24 format)

Choose the device that is appropriate for the media on which you have stored the backed up file system.

The interface then asks you to which directory you want to restore your data. You can select to restore data in a specific directory by giving the path of that directory. Then, later in this procedure when you are in interactive restore mode, you can restore a specific file in that directory. For example, assume the user *waltt* accidently destroyed the file *term.paper* in the directory */usr/waltt/Physics*. You can restore his file *term.paper* (later in the procedure when you are in interactive restore mode) by specifying in response to this prompt that you want to restore data to the directory */usr/waltt/Physics*.

If you want to restore the entire filesystem, press <RETURN> to use the default value of /.

If restoring from media, make sure that the first volume of the media you are restoring from is in the appropriate drive.

If you are restoring from cartridge tape, the sysadmin interface tries to read from */dev/tc*, so be sure you have set up this default tape drive link in */dev*. See Section 5 for details.

Once you have entered all the parameters, the interface displays the information you have entered, giving you one last chance to abort the restore if the information is not correct. You can abort the restore by pressing <ESC>. If the information is correct, press <RETURN> to execute the restore.

Interactive Restore Mode

After you answer all questions and press <RETURN> to begin the restore, the interface puts you into the *interactive restore* mode, which has its own set of commands. This is the same thing as entering the command **restore i**. The purpose of this mode is to let you put files and directories onto an *extraction list*, and then *extract* those files and devices from the restore media and put them onto the Winchester disk in the proper place in the file system. Put the first volume of your set of backup media in the appropriate drive after you enter the file system to restore, as this volume holds the directory of what is in this set of backup media.

The following are the commands that you can use to interactively restore one, some, or all files in a file system. These commands act on the directories on the backup media:

- ls [*dir*] — List the current directory, or the directory specified by *dir*. Entries in the list that are directories are followed by a /. Entries in the list that have been marked for extraction are preceded by a *. If the **verbose** key (see below) is set, the inode number of each entry is also listed.
- cd [*dir*] — Change the current working directory to *dir*.
- pwd — Print the full pathname of the current working directory.

- **add** [*arg*] — Add the current directory or specified argument to the list of files to be extracted. If no *arg* is specified, then the current directory, and all its subdirectories and their contents, are added to the extraction list. If a directory is specified as *arg*, then it and all its subdirectories and their contents are added to the extraction list. The easiest way to extract most of the files from a directory is to **add** the entire directory to the extraction list and then **delete** the files that you do not want to extract.
- **delete** [*arg*] — The current directory or specified argument is deleted from the list of files to be extracted. If no *arg* is specified, the current directory and all its subdirectories and their contents are removed from the extraction list. If a directory is specified as *arg*, then it and all its subdirectories and their contents are deleted from the extraction list.
- **extract** — All the files that are on the extraction list are extracted from the backup media and copied to the Winchester disk in the proper place in the file system. **Restore** asks which volume you want to mount. The fastest way to extract a few files is to start with the last volume in a set of backup media, and work towards the first volume.
- **verbose** — Toggles *verbose* mode. When set, verbose mode causes the **ls** command to list the inode numbers of all entries in the list. It also causes **restore** to print out information about each file as it is extracted.
- **help** — Lists the available commands.
- **quit** — **Restore** immediately exits, even if the extraction list is not empty. If you enter this command, you are returned to the menu level.

For more information on **restore**, see *restore(8)*. The information in *restore(8)* always calls the media you are restoring from a *tape*, even though you may be using diskettes.

Backup Over Network

This menu item runs the **rdump** command with the f and u options. The f option sents the dumped information to the device on the remote host you specify, and the u option is necessary for incremental dumps. For more information on these options, see the *rdump(8n)* manual page in the *UTek Command Reference* manual.

You cannot back up your system over the LAN unless you have the Network File System enabled. See the *Network File System Reference* manual for details on the Network File System.

When you choose this menu item, the sysadmin interface asks you to input a number of parameters. The default response to each prompt, which you can select by pressing <RETURN>, is displayed above the prompt. Items that the interface asks you for are:

- the level number (covered under Backup to Local Media)
- the hostname of the remote host
- your login name
- the filesystem you want to back up
- the remote device name

When the interface asks you to enter the name of the remote host, you must enter the name of a remote host where you have a personal account. Otherwise, you cannot back up data to that host. If for some reason network communications between your workstation and the remote host are not set up properly, backing up to the remote host won't work.

The interface asks you your login name, because (for security reasons) you cannot connect to another host on the LAN if you are *root*. Enter the login name for the account you use when you are not performing system administration tasks.

The interface then asks you for the name of the file system you want to back up. There is only one file system standard on the 6130, which is /.

Finally, the interface asks you to specify the device on the remote host. You must specify the device with a device file from the */dev* directory on the remote host. For example, If you wanted to back up your data to a 61TC01 cartridge tape drive connected to another workstation, and the workstation has the SCSI board to support the tape drive installed in slot 5 of the remote host, you would enter /dev/tc64 (this assumes the drive is set to the default device number of 4).

Once you have entered all the parameters, the interface displays the information you have entered, giving you one last chance to abort the backup if the information is not correct. You can abort the backup by pressing <ESC>. If the information is correct, press <RETURN> to execute the backup over the network.

Restore Over Network

This menu item puts you into an interactive mode that lets you restore one, some, or all files in a file system from the media on a remote host, but is most useful for restoring one or some files. See the *rrestore(8)* manual page in the *UTek Command Reference* manual for more details on restoring the system from a remote host.

The first thing the interface asks you when you choose *Restore Over Network* is your host name. You must have a personal account on the remote host to restore data rom devices on that host. If for some reason network communications between your workstation and the remote host are not set up properly, restoring from the remote host won't work.

After asking you for the remote host name, the interface asks you for your login name. It does this because you cannot connect to another host on the LAN if you are root, for security reasons. Enter the login name for the account you use when you are not performing system administration tasks.

Next, the interface asks you for the name of the remote device. Choose the device that is appropriate for the media on which you have stored the backed-up file system. You must specify the device with a device file from the */dev* directory on the remote host. For example, if you want to restore from a 61TC01 cartridge tape drive connected to another workstation, and the workstation has the SCSI board to support the tape drive installed in slot 5 of the remote host, you would enter /dev/tc64 (this assumes the drive is set to the default device number of 4). You should make sure the first volume of the media you specify is mounted on the remote host before you continue the procedure.

The interface then asks you to which directory you want to restore your data. You can select to restore data in a specific directory by giving the path of that directory. Then, later in this procedure when you are in interactive restore mode, you can restore a specific file in that directory. For example, assume the user *waltt* accidently destroyed the file *term.paper* in the directory */usr/waltt/Physics*. You can restore his file *term.paper* (later in the procedure when you are in interactive restore mode) by specifying in response to this prompt that you want to restore data to the directory */usr/waltt/Physics*.

To restore this standard file system, you can type / or <RETURN> (since / is the default).

Once you have entered all the parameters, the interface displays the information you have entered, giving you one last chance to abort the restore if the information is not correct. You can abort the restore by pressing <ESC>. If the information is correct, press <RETURN> to execute the restore over the network.

After you answer all questions and press <RETURN> to begin the restore, the interface puts you into the *interactive restore* mode, which has its own set of commands. This is the same thing as entering the command **restore i**. The purpose of this mode is to let you put files and directories onto an *extraction list*, and then *extract* those files and devices from the restore media and put them onto the Winchester disk in the proper place in the file system. See the earlier discussion of interactive restore mode for more information.

INSTALLATION OF OPTIONAL SOFTWARE

NOTE

This menu item is used to install new or revised software packages. Do not use this menu item to restore backups of system files. There is another menu item for that (see the previous discussion). If you try to use this menu item to restore a file, it does not work and you receive an error message.

Revised or optional software is distributed to 6130 workstations on diskette or streaming cartridge tape. Every software package comes with an Installation manual that tells you any special things you need to know to install that software. You should consult the documentation that came with the software package before attempting to use the sysadmin interface to install the software.

This menu choice takes you through the steps necessary for installing optional software, software updates, or applications onto your system. You can also list the contents of a diskette or tape. Both of these tasks require that the data be on diskette or streaming cartridge tape in **cpio** format. **Cpio** is a UTek file transfer program.

Install Software

You can install software from many media types:

- diskette
- streaming cartridge tape (QIC-24 format)
- a file on the Winchester disk
- network

You can also install software from a file on the workstation.

The data that you want to install *must* be in **cpio** format. If it's not, an error occurs, and the software is not installed.

The interface asks you from which device you want to install the software, and gives you the above list to choose from. If you choose diskette or streaming cartridge tape, make sure that the media is inserted into the appropriate device before you enter the number of the option. If you are installing from diskette, put the first diskette into the diskette drive. Diskettes are labeled *1 of n, 2 of n* and so on, where *n* is the total number of diskettes. The command that performs the installation is sent immediately when you enter the media type, and if the media is not installed in the device, an error occurs.

If you choose the network to install software from, you have to choose the host on the network from which you want to copy the software. Then, you have to choose the media on that host that the software is on: diskette, 9-track tape, streaming cartridge tape (either format), or a file in **cpio** format.

Once you have chosen the source device for the software, the system begins the installation process. The first prompt asks you to verify that your source media has been loaded. Press <RETURN> (or any key but **q**) to continue. After a short wait, the installation program displays the names of the files as it moves them from the software source to the Winchester disk.

If you are doing a software update, an attempt is made to save files commonly modified by the user, or files where data is stored by programs.

When you do an reinstallation procedure, messages are displayed telling you what files have been saved.

If Your Software Source Is Diskettes

The system displays this message when it has finished transferring all the files from the first diskette:

```
No more data on /dev/rdf

To continue this installation, insert the next diskette
in the drive, then press <RETURN>.

To quit press <q> followed by <RETURN>.

To install software from a different file or device, type
the new name, then press <RETURN>.

(See the section on multi-volume archives in cpio(1) for
more information.)

—>
```

To continue to install software from diskettes, follow these steps:

1. Remove the first diskette from the diskette drive. The first diskette is labeled *1 of n*, where *n* is the total number of diskettes.
2. Insert the second diskette (labeled *2 of n*) into the diskette drive.
3. Press <RETURN>.

   The system begins to copy files from the second diskette onto your Winchester disk.
4. Each time this message appears, replace the diskette in the drive with the next diskette of the installation package.

You must complete the entire process and install all the diskettes. If you must quit:

1. Type **q** followed by <RETURN>.

   The system displays these messages:

   ```
   Session terminated by user

   ***** SOFTWARE INSTALLATION FAILED *****

   Hit <RETURN> to return to the menu.
   ```

2. Press <RETURN>. The system returns you to the sysadmin interface.

   Remember, this interruption means that the software installation has *not* been completed. To install the software at a later time, you must begin all over again.

When All Files Have Been Copied

The system displays a message telling you where it has placed the list of all the files it has copied (the bill of materials). Then it completes the installation process with this message:

```
***** SOFTWARE INSTALLATION COMPLETE *****
```

If one of these messages appears:

```
SOFTWARE EXTRACTION FAILED
```

```
SOFTWARE INSTALLATION FAILED
```

the installation has not been correctly completed.

Verify the Installation

To verify your software installation, run your newly-installed program from your own account (not from the *root* account).

If the software does not run as expected, perform the installation procedure again. If it still does not run, contact your Tektronix Field Office.

Back Up the System

When you have installed and verified your optional software, take a backup of your system. Information on backing up the system is available earlier in this section.

List Contents of cpio File

This option lets you see the contents of a diskette or cartridge tape if the data on it is in cpio format. You would want to do this to see what is stored on the media.

NOTE

If you want to list the contents of a diskette that is part of a larger package, you must insert and remove all diskettes in the package for the List option to work properly.

USER LOGIN ACCOUNT MAINTENANCE

Account maintenance includes adding accounts for new users to the system, changing information about current users, and deleting accounts of users who no longer require access to the system. You can also list or search for information about specific users.

Each time this interface is entered, the files */etc/passwd*, */etc/group*, and */etc/chfn* are read to provide you with the information this interface gives you.

List Usernames

This menu item gives you a list of the login names on the system. It does this by reading the first field of each line in the */etc/passwd* file. For example, when you select this menu item, the interface might list *root*, *sysadmin*, *daemon*, *cron*, *sys*, *dist*, *uucp*, *admin*, *mdqs*, *user*, *test*, *waltt*, *joeu* and *jennyh* as valid user names.

If there are too many user names to be displayed on one screen, the sysadmin interface displays the first screen and asks you if you want to see more. You can hit any key (except q) to continue paging.

If a user name has had any changes made to it during the current session, the user name is preceded by an asterisk.

Full User Information Listing

This menu item gives you a full informational listing on each user, taken from the fields in the */etc/passwd* file. This listing provides the login name, whether there is a password, the userid, the home directory, the login shell, and any personal information. Example 4-2 shows how the listing for the user *joeu* might appear.

```
13) Name: joeu       Password: yes        Userid: 217
Home directory: /usr/joeu          Login shell: /bin/csh
Personal info: Joe User;555-1212;03-861
Groups: bigproj smallproj mktg
```

Example 4-2. Sample Full User Information Listing.

If there are too many user names to be displayed on one screen, the sysadmin interface displays the first screen and asks you if you want to see more. You can hit any key (except **q**) to continue paging.

For more information about the fields in the listing, see the discussion of the *Change User Information or Add User* menu item, later in this section.

Delete a User

If a user no longer needs system access, you may want to delete the user's account to save space and for security reasons. Copy the user's files to diskette or cartridge tape (see the **cpio** discussion in the *6130 System User's Guide*, or the *cpio(1)* manual page) in case someone else needs the information in those files. Then use the *Delete a User* menu selection to delete the account.

When you select this menu item, the interface prompts you for the login name of the user you want to delete by printing:

```
Delete user:
```

When you type in the user's login name or entry number (which you can get through the *Full User Information Listing* menu item) and press <RETURN>, the interface:

- Edits the */etc/passwd* file to remove the user's entry.
- Removes the user's login name from any groups in the */etc/group* file.

If the user is transferring to another host on your network, the administrator of that host should first add the user to the host. The user can then transfer files using the **rcp** command or the distributed file system (if available). After the files are transferred, you can use the *Delete a User* menu item to delete the user from your workstation.

The deleted user's userid is never reassigned if you let **sysadmin** fill in default userids, so that new users don't inherit the deleted user's files.

Change User Information or Add User

Selecting this menu item lets you add users or change existing user information. You can either save information while you are in the interface (by using *S* or *s*), or when you exit the interface. If you exit the interface after making changes but without saving them, the interface asks you if you want to save the changes you have made.

When you save changes, the interface writes changes to the */etc/passwd* file or */etc/group* file based on the responses you gave to the prompts. Only files that have had their contents changed are written.

Much of the information in */etc/passwd* can be changed by the user without your help. The **chsh** command lets users change their login shell, and **chfn** lets them change the information in the personal information form. The **passwd** command lets users change their passwords. The only fields that regular users can't change are *Login name*, *Userid*, *Group name*, and *Home directory*.

When you select the *Change or Add User* menu item, the interface prompts you for the login name of the user you want to add or change by displaying:

```
Change or add user:
```

You can enter a number of things in response to this prompt. You can:

- enter an existing login name
- the login name for a user who does not have an account, but for whom you want to create an account
- the entry number of an existing user (from the *Full User Information Listing*)
- the character @, followed by the login name of an existing users entry (this causes another prompt, which asks for the login name to be used for the entry. A copy is then made of the entry for the login name, and you edit the copy to create a new account.)

If you enter the name of a new user, remember that login names must be unique; that is, only one of each login name is allowed in the */etc/passwd* file (and therefore on the workstation). If the user is able to log in on more than one host in a network, the login name should be unique in the network, too. You can ask what each user prefers as a login name, or you can arbitrarily assign names. Login names are often the first name of the user followed by the first letter of the user's last name, without a space in between (for example, you might assign the login name *billg* for user Bill Goodguy). If there are conflicts, add another letter from the last name (for example, *billgo* instead of *billg*). The name can be from one to eight characters long.

When you have given the user a name in response to the prompt, a menu with three items is displayed in the menu space. These items let you *Edit Account Information*, *Edit Personal Information*, or *Edit Group Information*. Also, information about the user you specified is displayed in the workspace of the screen (in the same format as the *Full User Information Listing*; refer back to Example 4-1). If you typed in the login name of a user who already has an account, the interface displays the field names with the values for that user. If the login name is new, the fields are blank.

The fields displayed in the workspace are:

- Name
- Password
- Userid
- Home directory
- Login shell
- Personal info
- Groups

You can use the three menu items to change the information in any of these fields. The *Edit Account Information* selection lets you edit the *Name*, *Password*, *Userid*, *Home directory*, and *Login Shell* fields. The *Edit Personal Information* selection lets you specify the user's full name, their telephone extension, mail stop, home phone, and home computer. The *Edit Group Information* selection lets you specify the groups that you want the user to belong to (however, the group must exist before you can add a member to it; see the discussion on Group Account Maintenance for more information). Each of these menu items provides you with forms to fill in.

Guidelines for Entering Information

As you enter information about a user, follow these guidelines:

Login

This is the name that the user uses to log into the system. As explained earlier, the login name must be unique and be eight or less characters long.

Password

Users can choose their own passwords using the **passwd** command. If you are adding a new user or other type of account, you can:

- leave the password field empty when you are editing account information, which leaves the default of no password for the account
- enter **no login**, which means that only root can access the account
- assign the same password to all new users and instruct them to change it to one of their choice the first time they log in

The last method is preferable, so that you are not left with an unprotected account if the user neglects to enter a password. Good default passwords are *password*, *passwd*, or any other word you feel would be relatively easy to remember. The password you enter should be between 6 and 10 printable characters long.

Try to impress upon new users the importance of changing their passwords to something private. If they don't, then you can't give them a secure system.

When you enter a password using the *Edit Account Information* menu selection, the password is immediatedly changed to either *yes* (if you assigned a normal password) or *no login* after you press the <RETURN> key (if you entered *no login* as the password, you can only login from the root account). This prevents someone from entering the sysadmin interface and discovering the user passwords once they have been assigned.

Userid

The userid should also be unique to the host and the network for security reasons. The userid is a 5-digit field (maximum 32767). If you are setting up a new user's account, the userid defaults to the next consecutive unused number. If a user is deleted from the system, the userid is not automatically reassigned (but you can reassign the number by entering it in this field).

Also, when you use *Edit Account Information*, you can assign a userid by typing *@username*. When you do this, the userid for the given name is used.

If you are on a LAN, you might want to have a range of userids special to your system, so your userids do not conflict with userids on other systems. If you are on a network, either see your network administrator for a range of userids, or, if there is no network administrator, get together with other workstation owners on your LAN and assign ranges.

Home directory

This field should contain the directory into which the system puts the user at login. If you are setting up a new user account, you might want to enter */usr/login_name* as the user's home directory, where *login_name* is the new user's login name. This is the traditional home directory for a new user. If you want another directory to be the user's home directory, enter that directory in this field.

More than one user can have the same home directory.

Login shell

This field should contain the full pathname of the shell that the user enters upon logging in. The Bourne Shell (*/bin/sh*) is the default shell. When editing account information, you can leave the */bin/sh* in this field, or enter some other login shell. If you want the C-shell (available with the UTek/A software package) as the login shell, enter **/bin/csh** in this field. Bourne shell and C-shell are the only two shells shells available with this release of UTek. If you install or develop another shell and want to use it, enter the full pathname of the file where the shell is stored.

Group name

It is usually a good idea to assign users who are working on the same or similar projects to the same group. If you have not defined groups using the *Group Account Maintenance* function of the interface, *gen* is the user's default group. You must define a group before you can add a user to it. The sysadmin interface will not let you add a member to a nonexistent group.

If, when you use *Group Account Maintenance*, you try to enter the name of a group that has not yet been defined, the interface asks you if you want the group added to the */etc/group* file. (Note that the group name is translated into a groupid in the */etc/passwd* file.) Group names can be up to 10 characters long. Groups are discussed in greater detail later in this section.

Edit Account Information

This form lets you add or change a user's login name, password, userid, home directory or shell. When you select this menu item, a form is displayed containing the existing values (or default values if you are creating a new user) for each field. In this example, the default values are shown in parentheses:

```
Username: joeu
Password: yes (no)
Userid  : 217 (next available)
Home    : /usr/joeu (/usr/name)
Shell   : /bin/csh (/bin/sh)
```

The interface places the cursor at the beginning of the value in *Username* field.

If you want to go to the next field, use the <CR> or <TAB> keys. When you get to the last field on the form, pressing <CR> or <TAB> moves you back to the first field again. To go back to a previous field, press <CTRL-B>. To get back to the main menu, use the <ESC> key.

Edit Personal Information

NOTE

You do not have to fill out the personal information for every user from the sysadmin interface. Users can fill in their own personal information using the **chfn** *command.*

The personal information form contains the information that goes into the fifth field of the */etc/passwd* file (separated by semicolons). This is information like the user's full name, office extension, office number, and possibly home address and phone number. The fields in the personal information form are variable. You can set them by editing the */usr/lib/chfn* file. Then, the fields you set appear each time you enter the personal information form.

When you select this menu item, the form is displayed containing the existing values (or no values if you have never specified personal information for this user) for each field:

```
Name                              : Joe User
Extension (Example: 682-347Ø)     : 555-1212
Mail Stop (Example: 61-261)       : Ø3-861
Home Phone                        :
Machine                           :
```

The interface places the cursor at the beginning of the value in the first field.

If you want to go to the next field, use the <CR> or <TAB> keys. When you get to the last field on the form, pressing <CR> or <TAB> moves you back to the first field again. To go back to a previous field, press <CTRL-B>. To get back to the main menu, use the <ESC> key. If you want to delete the information in a field, move to that field using <CTRL-B>, <CR>, or <TAB> and then press <CTRL-X>.

The chfn File The */usr/lib/chfn* file contains fields that define the personal information form. This file also defines the questions asked by the **chfn** utility. See *chfn(1)* for details on the **chfn** utility.

By editing the *chfn* file, you can set the fields in the personal information form. Each line in the file represents a field in the form.

The field label should begin the line, followed by a semicolon (;). Anything after the semicolon should indicate the character or range of characters that you should enter into the field. Example 4-3 shows a sample *chfn* file. Note that ranges of characters are represented within [square brackets]. See *chfn(5)* for details on the */usr/lib/chfn* file.

```
Name;
Extension (Example: 682-3470);^[1-9][0-9][0-9]-*[0-9][0-9][0-9][0-9]$
Mail Stop (Example: 61-215);^[Y0-9][0-9]-[0-9][0-9][0-9]$
Home Phone;^[1-9][0-9][0-9]-*[0-9][0-9][0-9][0-9]$
```

Example 4-3. Sample chfn File.

Edit Group Information

Selecting this menu item shows you the groups that the user is in. You can then add groups, delete groups, or change existing groups. The information you enter is added to the */etc/groups* file when you exit the interface. However, the group must exist before you can add a member to it.

If you want to go to the next field, use the <CR> or <TAB> keys. When you get to the last field, pressing <CR> or <TAB> moves you back to the first field again. To go back to a previous field, press <CTRL-B>. To get back to the main menu, use the <ESC> key. If you want to delete the information in a field, move to that field using <CTRL-B>, <CR>, or <TAB> and then press <CTRL-X>.

You can list all the group names in */etc/groups* by pressing <CTRL-G>. You can get a full listing of the groups (find out who is in a group) by pressing <CTRL-F>. This full listing is discussed in the next heading under Group Account Maintenance.

Saving Changes

Once you have made all the changes to the user accounts that you want, type S or s to save the changes in the */etc/passwd* file and return to the top level Sysadmin Menu.

GROUP ACCOUNT MAINTENANCE

The Group Accounting menu appears when you choose this menu item. The Group Accounting menu lets you list, add, delete, change group information stored in the */etc/group* and */etc/passwd* files, and globally delete a user in the */etc/group* file. Details on the information in the */etc/group* file are discussed in Section 6 under Users and Groups.

Once you have made a change, the interface displays a message that tells whether messages have been made to the group data, the password data, neither, or both.

List Group Information

This item lists all the groups currently in */etc/group*. Each numbered item on the list includes the group's name, its groupid, and the members of the group. Example 4-4 shows possible output generated by this selection:

```
      Currently defined groups

1) Name: other          Id: 0
   Members:       root sysadmin

2) Name: daemon         Id: 1
   Members:       daemon cron uucp

3) Name: sys            Id: 2
   Members:       admin dist sys mdqs

4) Name: gen            Id: 20
   Members:       test user bandit davidl pamd marcw
                  karinw keithl ariels waltt
```

Example 4-4. List of Groups.

Add a Group

When you choose this selection, the following form appears in the workspace:

```
Adding new group (Enter <ESC> to abort addition)

Group name       :
Group id number  :
Group member     :
```

Fill in this form with the following information to add a group to */etc/group*:

Group name — This name must be unique on the workstation. The name may be from one to eight alphanumeric characters long, and the - (hyphen) and _ (underscore) are also valid characters. The group name must start with a letter.

Group id number — This number should be unique on the workstation. Choose an integer in the range between 0 and 32767. You may want to set ranges of groupids for two reasons:

- For the network, so that groupid numbers do not overlap between machines unless the groups correspond to each other over the network.
- Within the range of groupids on your workstation, so that you can assign subgroups. For example, you can assign the engineering group groupids 200-230, with group 200 containing all engineers. Then later, you can create project-related groups containing a subset of engineers and still keep the engineering groupids close to each other.

If you press <RETURN> when you are in this field, the groupid defaults to the next unused groupid number (this is determined by adding 1 to the highest groupid number on the workstation).

Group member — Enter the login names of the users you want to put in the group one at a time. Press <RETURN> after each login name you type. As you type in each login name, it is added to a list of group members that appears under the form in the workspace. When you have added all the users to the group, press <RETURN> when the Group member field is empty. This causes the new group to be saved in the group data.

Delete a Group

This item lets you delete a group. The entry for the group is deleted from the */etc/group* file.

The following line appears in the workspace:

```
Delete group:
```

Type in either the name of the group you want to delete, or the list number that is associated with the group when you choose the *List Group Information* menu item and press <RETURN>. If you decide you do not want to delete a group, press <RETURN> without filling in a group's name or list number.

Change a Group

You can add users to an existing group, delete users from a group, change the group's name or groupid with this menu selection.

When the line `Change group:` appears on the screen, type in either the group's name, or its list number from the *List Group Information* menu item and press <RETURN>. If you decide that you do not want to change any group, press <RETURN> without filling in a group's name or list number.

Once you select the group to change, a new menu appears in the menu space, and a form listing the group's name, groupid, and membership appears in the workspace. As you make changes to the group, the form in the workspace reflects the changes.

Change Group Name

This item lets you choose a new group name. The rules for selecting a group name discussed earlier under Add a Group apply.

Change Groupid Number

This item lets you choose a new groupid. The rules for selecting a groupid discussed earlier under Add a Group apply.

Add a Group Member

This item lets you add a login name to the list of group members. If you want to add more than one login name to the list, choose this item again for each member you want to add.

Delete a Group Member

This item lets you delete a user from a list of group members. If you want to delete more than one group member, choose this item again for each member you want to delete.

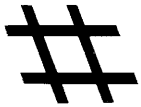
Globally Delete a Group Member

This item lets you delete a user from all groups except the user's default group. This is useful if you are removing a user from the system.

When the line `Delete member` appears in the workspace, type in the login name you want deleted from all groups and press <RETURN>. If you decide that you do not want to globally delete a user, press <RETURN> without filling in a login name. If you want to globally delete more than one user, choose this item again for each user you want to delete.

Saving Changes

Once you have made all the changes to groups that you want, type S or s to save the changes in the */etc/group* file and return to the top level Sysadmin Menu.



5

Other Procedures

Introduction

This section covers the common system operations that cannot be handled through the sysadmin interface. These procedures include:

- System start–up
- Adding devices
- Formatting diskettes
- Setting the time and date
- System shutdown

This section also contains information on how to use the **restore** command to restore the entire system.

This section assumes that you understand some things about UTek:

- The basic file structure of UTek and how to move about in it.
- The concepts of file protections and file ownership.
- How to use one of the system text editors (preferably vi).

You can find information on these subjects in *Introducing The UNIX System* by Henry McGilton and Rachel Morgan, in the *6130 Learning Guide*, and in the online sessions.

System Start–up

This discussion concentrates on the basic start–up procedure. If this is the first time you are starting the system, refer to Section 2, First Time Start–up.

The basic start–up procedure is automatic. When you turn the workstation on, the system:

- Goes through a number of diagnostic checks.
- Possibly runs a file system check (depending on how the system was shut down).
- Initializes daemon programs for the network and for local use.
- Brings the system up in multiuser (normal operating) mode; that is, displays the **login:** prompt.

Before you press the start/stop switch, you should:

1. Check the configuration switches on the workstation back panel to make sure that:
   a. The system is set to boot mode (Switch 1 is up).
   b. The appropriate console device is selected. (Switches 2 and 3 determine this. Table 5–1 shows the meanings of Switches 2 and 3).
   c. The system is set to boot UTek from the Winchester disk (Switch 4 is up).
   d. The appropriate boot device is selected. (Switches 5 and 6 determine this. Table 5–2 shows the meanings of Switches 5 and 6).

   Figure 5–1 shows the workstation back panel and the location of the Configuration switches on the back panel.

Table 5–1
CONSOLE DEVICE SETTINGS

Console Device	Switch 2	Switch 3
6100 Series Display	up	up
9600 baud RS–232–C terminal (port 1)	up	down
1200 baud RS–232–C modem/terminal (port 0)	down	up
300 baud modem/terminal (port 0)	down	down

2. Check that the yellow line voltage indicator on the workstation back panel (see Figure 5–1) shows the correct voltage. Choices are 110 Vac (domestic), or 220 Vac (European).

NOTE

Changing the position of this indicator does not change the line voltage. If you need to change the line voltage, contact your local Tektronix Field Office.

Table 5–2
BOOT DEVICE SETTINGS

Boot Device	Switch 5	Switch 6
Autoboot	up	up
Winchester disk	up	down
Diskette drive	down	up
LAN port	down	up

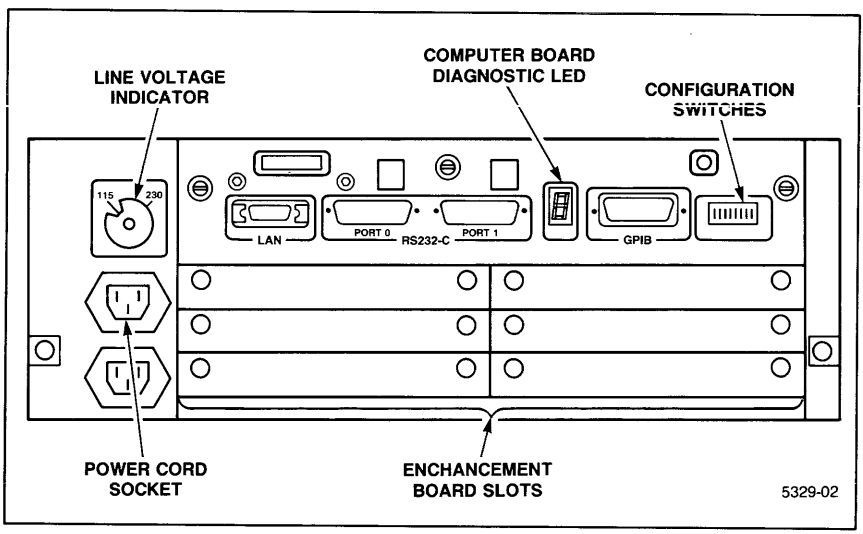


Figure 5–1. 6130 Workstation Back Panel.

3. Turn on the console device (selected by configuration switches 2 and 3, see Table 5–1). If you don't know which terminal is the console device, turn them all on. The terminal that displays the diagnostic messages is the console.
4. Turn on any external peripherals such as extra terminals, printers, or plotters.
5. If you want to boot the system from diskette, insert the diskette you want to boot from (generally, the miniroot system diskette from the diskette distribution) into the diskette drive. If configuration switches 5 and 6 are set to autoboot (see Table 5–2), the system attempts to boot first from any diskette in the diskette drive.
6. Press the start/stop switch located in the lower right corner of the workstation front panel. The green light on the switch should go on.

At this point, the start–up diagnostics should begin. If the green light doesn't go on, or the diagnostics don't begin, refer to Section 8, System Halts.

Start-up Diagnostics

When you press the start/stop switch, the workstation goes through a number of ROM diagnostic tests before you see anything on the console screen. These diagnostic tests are loaded from the workstation's read-only memory (ROM). As each test is executed its test number is displayed on the seven-segment Computer Board Diagnostic LED, located on the workstation's back panel (see Figure 5-1). If a test fails, the test's number remains on the LED. If this type of failure occurs, see the *6130 Diagnostics* manual or contact your Tektronix Field Office. If all tests complete successfully, the seven-segment LED turns off momentarily.

At this point, the workstation automatically runs power-up diagnostics. These diagnostics are part of the diagnostic operating system. They are loaded from the Winchester disk and executed. The seven segments of the LED flash in a race track pattern.

Diagnostic messages may appear on the console screen if there are nonfatal errors. If a fatal hardware error occurs during these tests, the LED panel on the back of the workstation lights in a pattern that indicates the error. Figure 5-1 shows the location of the LED panel on the back panel of the workstation. If either of these occur, see the *6130 Diagnostics* manual or contact your Tektronix Field Office.

During these tests, if the settings of the configuration switches have been changed since the last time the system was started the following message appears:

```
System configuration has changed since last boot
Update config file to reflect new configuration? [y,n,(y)]
```

Always answer **y** to this question. If you just press ⟨RETURN⟩, the answer defaults to "yes".

After you answer this question, the diagnostic tests start over again and run all the way through.

When the diagnostic operating system is finished running tests, the LED turns off momentarily.

If no errors occur during the power-up diagnostics, the workstation boot program begins. Messages from the boot program appear on the screen. These messages include a list of the devices the boot program finds, memory allocation data, and so on. Example 5-1 shows a typical list of these boot messages.

```
Firmware Version 02

beginning boot program execution

  ************************************************************
  *                                                          *
  *                      TEKTRONIX                           *
  *                                                          *
  *       6100 SERIES INTELLIGENT GRAPHICS WORKSTATION       *
  *                                                          *
  ************************************************************

scanning option slots
testing computer board
testing 1/2 M option memory in slot 2
mapping caches and expansion memory
booting /vmunix
        RSA - RS-232 ports
        DW - Hard Disk drive
        GPA - GPIB port
        lna0: Ethernet= 8:0:11:0:80:9 Ip=[7.0.80.9]
        LNA - Local Area Network
        CEPWR - Soft power switch
        DF - Floppy Disk drive
end configure
Tektronix - UTek RCS #1.84 Wed Oct 31 10:14:24 PST 1984

real mem = 786k
firstaddr = 0x3b718
firstaddr = 0x5d200
avail mem = 379904
using 38 buffers containing 77824 bytes of memory
```

Example 5-1. Sample Boot Program Messages.

The boot program then checks to see if the */fastboot* file exists. If the file exists, then the file system check (**fsck**) performed at the last shutdown ended with a healthy file system, so the system does not run **fsck** now.

If the system doesn't run **fsck** at boot, the following message appears on the console screen:

```
Fast boot ... skipping disk checks
```

If the boot program doesn't find the */fastboot* file, it starts up **fsck** in *preen* mode. You can tell that the system is running **fsck** if the following message appears on the console screen:

```
Automatic reboot in progress . . . date
```

where *date* is the date and time according to the workstation's internal clock.

If the system runs **fsck**, the boot procedure takes a few minutes longer than if the system doesn't run **fsck**. You know that **fsck** is finished when two lines similar to the following appear on screen:

```
/dev/dw00a: 1061 files, 9591 used, 3520 free (160 frags, 420 blocks)
Mon Aug  6 08:59:59 PDT 1984
```

Anything appearing between the "Automatic reboot" line and the "/dev/dw00a" line is an **fsck** message. For details on **fsck**, see Section 7, and for a list of **fsck** messages and explanations, see Appendix A.

If the boot **fsck** finds problems that it can't fix in *preen* mode, the system comes up in single user mode. This happens very rarely. If you see the # prompt here, you know that you are in single user mode, and you should run **fsck** in interactive mode. To run **fsck** on the root file system (the only file system that comes standard on the 6130), type **fsck**. For more information on **fsck**, see Section 7. For a list of **fsck** messages, see Appendix A.

After the **fsck**, or after the **Fast boot** message if there's no **fsck**, the system goes into multiuser (normal operating) mode and the system daemons start. Finally, the **login:** prompt appears on the screen.

Adding Devices

You may want to add external devices to your workstation. The following discussion deals with the devices that are standard to the 6130, the devices you can add to the 6130, and the procedure for adding new devices to the UTek system.

The MAKEDEV Utility

The **MAKEDEV** utility creates device files in the */dev* directory when you give it the correct parameters. It knows about the drivers in the kernel, and when you give it the correct parameters, creates a device file with the right name and links that file to the appropriate driver.

MAKEDEV is not the only way to create device files; you can use the **mknod** utility. However, **MAKEDEV** is the easiest way to create devices and is therefore the only method this manual discusses. See *MAKEDEV(8)* for a detailed discussion of the command.

If you are adding a device that is not in the standard kernel, you must run **sysconf** to reconfigure the kernel. **Sysconf** also creates a new version of **MAKEDEV** that you must copy to */dev/MAKEDEV*. See Section 9 for details on **sysconf**.

Making New Devices

To make a new device file, the procedure is basically the same as remaking the standard devices. The difference is in the argument you specify for the **MAKEDEV** command.

Before you make a device file for the enhancement device, you should install any hardware (boards, connectors, cables) that the device requires. See the *6130 Installation Guide* enhancement inserts for details on installing hardware for enhancements. Remember the slot number (1 through 6) of the backplane slot where you install the enhancement board, as you need this number to configure the system software to deal with the enhancement. Figure 5–2 shows the numbers of the backplane slots. Table 5–3 shows the recommended slots for enhancement boards. A half–size board takes up one slot. A full size board requires two slots.

Table 5–3
6130 ENHANCEMENTS

Product	Description	Size	Slot Location
61MP01	512 kbyte Memory Expansion	Full	1 and 2
61MP02	1 Mbyte Memory Expansion	Full	1 and 2
61MP03	2 Mbyte Memory Expansion	Full	1 and 2
61KP01	Hard Copy Interface	Half	5 or 6
61KP03	High Speed GPIB	Half	5 or 6
61KP04	SCSI Interface (necessary for the 61TC01)	Half	6
61KR01	Dual RS–232–C Interface	Half	5 or 6
61KR02	Sync/Async Interface	Half	5 or 6

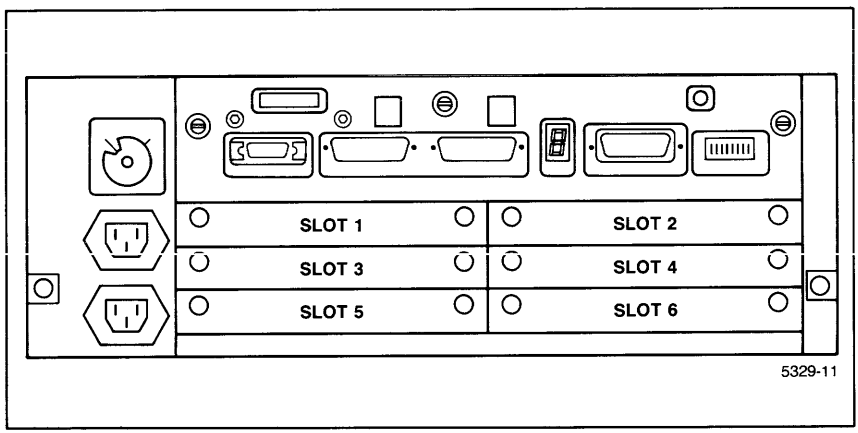


Figure 5–2. Backplane Slot Numbers.

No matter what device you are installing, the procedure begins the same way:

1. Install the hardware enhancement according to the instructions included with the enhancement.
2. Turn on the workstation. It should come up in normal, multiuser mode.
3. Log in as *root*.
4. If the enhancement came with diskettes, install the software on these diskettes using the *sysadmin* interface. Some enhancement products may come with diskettes that contain the device driver and diagnostics for the enhancement. You must install the contents of these diskettes the same as if you were installing optional software (see the installation manual that came with the enhancement or Section 4 of this manual for details).
5. Install the **sysconf** kernel reconfiguration package if it is not already installed. **Sysconf** is contained on four of the nine diskettes that you received with your 6130 workstation and can be installed using the *sysadmin* interface. See section 9 for more directions on installing **sysconf**.
6. Build a new kernel using the **sysconf** software. The new kernel contains the device driver for the new device. See section 9 for directions on how to build a new kernel using **sysconf**.
7. Replace of the old kernel (contained in the file */vmunix*) with the new kernel (located in */usr/sys/conf/vmunix*) you just created.
   a. Change the directory to the root directory by typing:

      ```
      cd /
      ```

   b. make a back up copy of the old kernel by typing:

      ```
      mv /vmunix /vmunix.old
      ```

      This is a safety measure to make sure you still have the old kernel until you have sucessfully booted the new kernel.

   c. Move the new kernel to the root directory by typing:

      ```
      cp /usr/sys/conf/vmunix /vmunix
      ```

8. Replace the old file */dev/MAKEDEV* with the new */usr/sys/conf/MAKEDEV* file you just created using **sysconf**.

   a. Make a back up copy of the *MAKEDEV* script by typing:

      ```
      cp /dev/MAKEDEV /dev/MAKEDEV.old
      ```

   b. Move the new *MAKEDEV* script into the */dev* directory by typing:

      ```
      mv /usr/sys/conf/MAKEDEV /dev/MAKEDEV
      ```

9. Use the **shutdown** command to bring the system down to single user mode. (See the discussion of the **shutdown** command later in this section.)

10. Change your current directory to */dev* by typing:

    ```
    cd /dev
    ```

Refer to the part of following discussion that corresponds to the device you are installing.

If you are installing more than one device, do the hardware installation for all the devices first, then do the software installation for all the device files before returning the system to multiuser (normal operating) mode.

Dual Hard Copy Interface

If you are installing a Dual Hard Copy interface, you can choose from four different drivers when you create a device file. Peripherals that connect to this interface should be Centronics-compatible.

The device files that you can create have one of the following four forms:

/dev/hcsp This device driver (*hc*) assumes that the device connected to the port behaves like a Tektronix printer. The *s* specifies the slot number (1 through 6, see Figure 5–2) where you installed the Dual Hard Copy enhancement board. The *p* specifies the port number you are creating the device file for (0 or 1).

The command to create this device file is:

MAKEDEV hc*sp*

/dev/uhcsp This device driver (*uhc*) converts lowercase letters to uppercase letters. The *s* specifies the slot number (1 through 6, see Figure 5–2) where you installed the Dual Hard Copy Interface. The *p* specifies the port number you are creating the device file for (0 or 1).

The command to create this device file is:

MAKEDEV uhc*sp*

/dev/chcsp This device driver (*chc*) converts linefeed characters to carriage-return/linefeeds. The *s* specifies the slot number (1 through 6, see Figure 5–2) where you installed the Dual Hard Copy enhancement board. The *p* specifies the port number you are creating the device file for (0 or 1).

The command to create this device file is:

MAKEDEV chc*sp*

/dev/uchcsp This device driver (*uchc*) converts lowercase to uppercase *and* converts linefeed to carriage-return/linefeed. The *s* specifies the slot number (1 through 6, see Figure 5–2) where you installed the Dual Hard Copy enhancement board. The *p* specifies the port number you are creating the device file for (0 or 1).

The command to create this device file is:

MAKEDEV uchc*sp*

If you plan to attach a printer to either of the ports, you should use the Spooler Configuration (MDQS) capability of the sysadmin interface to set up a print queue for the printer(s).

Once the device files are created and you have set up any queues, if these are the only device files you are creating, or they are the last ones you have to create, type ⟨CTRL–D⟩ to return the system to multiuser mode. This restarts system daemons and returns the system to normal operation.

If you have other devices to install, do that before returning to multiuser mode.

Optional Dual RS-232-C Interface

If you are installing an optional Dual RS–232–C interface, enter the command:

MAKEDEV tty*s*0 tty*s*1

Where *s* is the slot number (1 through 6, see Figure 5–2) where you installed the Dual RS–232–C interface board. This creates devices:

- */dev/ttys0*
- */dev/ttys1*

Where *s* is the slot number (1 through 6, see Figure 5–2) of the enhancement board.

After you create these device files, use the Port Configuration feature of the sysadmin interface to set the login status (yes or no) of the ports, and to assign terminal types and baudrates. If you plan to connect a printer or plotter to one or both ports, use the MDQS Configuration feature of the sysadmin interface to set up a queue for the device you plan to connect.

Once the device files are created and you have set up any queues, if these are the only device files you are creating, or they are the last ones you have to create, type ⟨CTRL–D⟩ to return the system to multiuser mode. This restarts system daemons and returns the system to normal operation.

If you have other devices to install, do so before returning to multiuser mode.

Optional GPIB Ports

If you are installing the optional GPIB interface, enter the command:

MAKEDEV gpib*s*

Where *s* is the slot number (1 through 6, see Figure 5–2) where you installed the GPIB board. This creates devices:

- */dev/gpibs*
- */dev/gpids*

Where *s* is the slot number (1 through 6, see Figure 5–2) of the enhancement board.

Remember that the *gpib* device file represents the device that users access, and *gpid* is used only by configuration software.

Once the device files are created and you have set up any queues, if these are the only device files you are creating, or they are the last ones you have to create, type ⟨CTRL–D⟩ to return the system to multiuser mode. This restarts system daemons and returns the system to normal operation.

If you have other devices to install, do that before returning to multiuser mode.

More GPIB devices can be created by regular users. They, and the UTek GPIB specific commands, are discussed in greater detail in the *6130 System User's Guide*.

SCSI Interfaces

A SCSI enhancement board is available for the 6130. *SCSI* stands for Small Computer Systems Interface. The SCSI is used to connect to mass storage devices, such as a streaming cartridge tape drive or an external Winchester hard disk, that contain a SCSI interface. If you plan to connect a 61TC01 streaming cartridge tape drive to your workstation, you need to have the 61KP04 SCSI enhancement board. The 61KP04 board should be installed in slot six to use the factory default values for SCSI devices.

You do not have to run **MAKEDEV** for the SCSI until you attach a device controller to the interface. In fact, you cannot specify a device number (*d*) to the **MAKEDEV** command until you know to which of the eight SCSI addresses (0 through 7) the controller will be connected. The SCSI enhancement board uses address 7.

Calculating Device Numbers

Device numbers for devices on the SCSI depend on the address of the device controller on the interface, and the number of the device unit on the controller. The equation for calculating the device number is:

device number *d* = (2 * controller location) + unit number

Since the device number must be a single digit, device numbers greater than 9 become uppercase letters: 10 becomes **A**, 11 becomes **B**, and so on.

The device number for all Tektronix 61TC01 Streaming Cartridge Tape Drives is set to 4 at the factory. (The default controller number is 2, and the default unit number is 0, so using the above equation (2 * 2) + 0 = 4). The If you are installing only one 61TC01 tape drive, leave the device number at 4.

If you are installing more than one 61TC01 tape drive, only one of them may have the factory default device number of 4. You must change the device numbers on the other 61TC01 tape drive(s) so that each has a unique device number. For information on changing drive numbers, see the *61TC01 Streaming Cartridge Tape Drive Instruction Manual* (this manual may refer to device numbers as controller numbers or device addresses). Use the numbers you set as the controller location in the equation for the device number in the *MAKEDEV* command for these tape drives.

Streaming Cartridge Tape

NOTE

The procedure for **MAKEDEV** *is slightly different if the 61TC01 contains the option 14 or 15 hard disk drive.* **MAKEDEV** *treats the hard disk drive and the streaming tape drive as two separate devices, even though they share the same physical cabinet. However, if you have a 61TC01 with a hard disk drive, you can make the devices for both the streaming tape and the disk drive at the same time. For more information on the 61TC01 with a hard disk drive, see the discussion following this one.*

The **MAKEDEV** utility for the streaming cartridge tape creates two different device files. They are:

/dev/tcsd — This driver (*tc*) signals the tape drive to rewind after the tape access operation, and expects the tape to be in standard (QIC–24) density format. The *s* specifies the slot where you installed the SCSI extension board (1 through 6, see Figure 5–2). The *d* specifies the device number for the device. (See the earlier discussion on calculating device numbers).

/dev/ntcsd — This driver (*ntc*) does not signal the tape drive to rewind after the tape access operation, and expects the tape to be in standard (QIC–24) density format. The *s* specifies the slot where you installed the SCSI extension board (1 through 6, see Figure 5–2). The *d* specifies the device number for the device. (See the earlier discussion on calculating device numbers).

To create these two device files, enter the command:

MAKEDEV tc*sd*

Where *s* is the slot number where you installed the SCSI enhancement board (1 through 6, see Figure 5–2), and *d* is the device number of the device on the SCSI (see the earlier discussion on calculating device numbers).

If you want the streaming cartridge tape drive you just installed to be the *default streaming cartridge tape drive*, you must specify this in the **MAKEDEV** command line. The default streaming cartridge tape drive is the drive that the backup, restore, and installation features of the sysadmin interface send data to and read data from.

To specify that this drive is to be the default drive, substitute the following command line for the previous one:

MAKEDEV tc*sd* **tc**

Where *s* is the slot number where you installed the SCSI enhancement board (1 through 6, see Figure 5–2), and *d* is the device number of the device on the SCSI (see the earlier discussion on calculating device numbers).

This creates links from the device files that the sysadmin interface calls to the device files you are creating for the streaming cartridge tape drive.

Once the device files are created and you have set up any queues, if these are the only device files you are creating, or they are the last ones you have to create, press ⟨CTRL–D⟩ to return the system to multiuser mode. This restarts system daemons and returns the system to normal operation.

If you have other devices to install, do so before returning to multiuser mode.

61TC01 Hard Disk Drive

If you have a 61TC01 with the option 14 or 15 hard disk drive, **MAKEDEV** treats the hard disk as a separate device. The **MAKEDEV** utility for the 61TC01 hard disk drive creates the files for the streaming cartridge tape, plus two different device files for the hard disk.

The files for the streaming tape are:

/dev/tcsd This driver (*tc*) signals the tape drive to rewind after the tape access operation, and expects the tape to be in standard (QIC–24) density format. The *s* specifies the slot where you installed the SCSI extension board (1 through 6, see Figure 5–2). The *d* specifies the device number for the device. (See the earlier discussion on calculating device numbers).

/dev/ntcsd This driver (*ntc*) does not signal the tape drive to rewind after the tape access operation, and expects the tape to be in standard (QIC–24) density format. The *s* specifies the slot where you installed the SCSI extension board (1 through 6, see Figure 5–2). The *d* specifies the device number for the device. (See the earlier discussion on calculating device numbers).

The files for the option 14 or 15 hard disk are:

*/dev/ds*sd*[a,b,l]* These drivers (*ds*) specify block buffered data for specific partitions of the disk (see Figure 5–3). Block buffered data is used for normal operations such as **mount** and **unmount**. At the end of the device file is either a *a, b,* or *l*, which specifies the disk partition to write to or read from. Partition *a* is the file system. Partition *b* is the swap space. Partition *l* is both the *a* and *b* partitions together. Like the device files for the streaming tape, the second *s* specifies the slot where you installed the SCSI extension board (1 through 6, refer back to Figure 5–2). The *d* specifies the device number for the device. (See the earlier discussion on calculating device numbers). For example, the three default entries for block buffered partitions are *ds66a*, *ds66b*, and *ds66l*. The *66* indicates that the SCSI board is in slot 6, and the hard disk drive is device number six.

*/dev/rds*sd*[a,b,l,p]* These drivers (*rds*) specify raw (unbuffered) data for specific partitions of the disk (see Figure 5–3). Raw data is used for *format*, **newfs**, **fsck**, and **dump** operations. At the end of the device file is either a *a, b, l* or *p*, which specifies the disk partition to write or read from. Partition *a* is the file system. Partition *b* is the swap space. Partition *l* is both the *a* and *b* partitions together (partition *l* is used by **newfs**, and is the default for **fsck** and **dump**. Partition *p*, which is not available for buffered disk partition files, is the entire disk, including various diagnostic areas and maintenance tables. Like the device files for the streaming tape, the second *s* specifies the slot where you installed the SCSI extension board (1 through 6, refer back to Figure 5–2). The *d* specifies the device number for the device. (See the earlier discussion on calculating device numbers). For example, the four default entries for raw partitions are *rds66a*, *rs66b*, *rds66l*, and *rds66p*. The *66* indicates that the SCSI board is in slot 6, and the hard disk drive is device number six.

Refering back to the device number equation, the default hard disk controller number is 3, and the default unit number is 0, giving the 61TC01 optional hard disk a default device number of 6.

61TC01 Cartridge and Hard Disk Drives

To create the device files for both the streaming tape and the 61TC01 optional hard disk drive, enter the command:

MAKEDEV tc*sd* **tc ds***sd*

Where *s* is the slot number where you installed the SCSI enhancement board (1 through 6, refer back to Figure 5–2), and *d* is the device number of the device on the SCSI (see the earlier discussion on calculating device numbers).

This creates links from the device files that the sysadmin interface calls to the device files you are creating for the streaming cartridge tape drive and the 61TC01 optional hard disk .

Once the device files are created and you have set up any queues, if these are the only device files you are creating, or they are the last ones you have to create, press ⟨CTRL–D⟩ to return the system to multiuser mode. This restarts system daemons and returns the system to normal operation.

If you have other devices to install, do so before returning to multiuser mode.

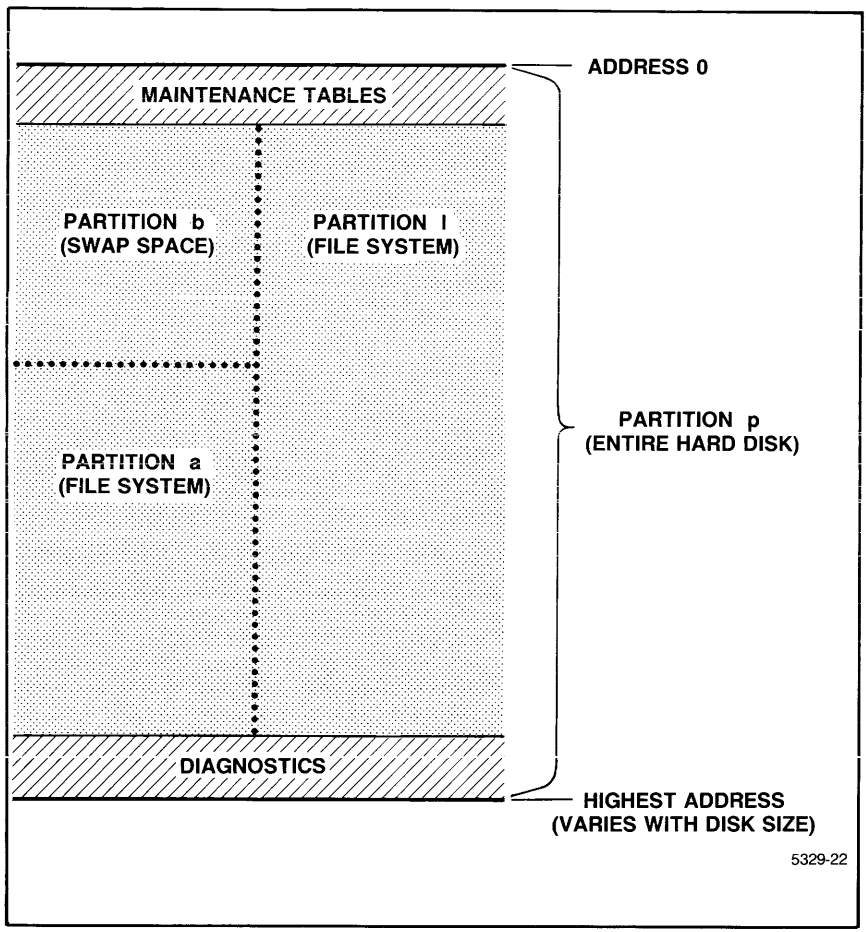


Figure 5-3. 61TC01 Optional Hard Disk Drive Partitions.

Remaking the Standard Devices

You should not have to make the standard device files (discussed earlier in this section) unless one or more of them is accidentally removed. Since this type of accident can happen, **MAKEDEV** has an option that automatically recreates all the standard device files discussed earlier.

Use the following procedure to remake the standard device files:

1. Log in as *root*.
2. Use the **shutdown** command to bring the system down to single user mode. (See the discussion of the **shutdown** command later in this section.)
3. Change your current directory to */dev* by typing:

   ```
   cd /dev
   ```

4. Remake the standard device files by typing:

   ```
   MAKEDEV std
   ```

5. When the # prompt comes back, return the system to multiuser (normal operating) mode by typing ⟨CTRL–D⟩. This restarts the system daemons. The **login:** prompt appears on the screen when the system is back in multiuser mode.

You can also use **MAKEDEV** to remake only some of the standard devices. However, it is better to use the **std** option and remake all the devices if only some need remaking than to use the more specific options. For more information on these other options for remaking standard devices, see *MAKEDEV(8)*.

Formatting Diskettes

Before you can write data onto a diskette, the diskette must be in the proper format. The program that formats a diskette is **saformat**, the same program that is used to format the hard disk.

Saformat is on the *standalone utilities* diskette of the diskette distribution. It doesn't run under UTek. You must shut the workstation down to run **saformat**. Therefore, it may be more convenient to format a group of diskettes at one time, then distribute them to users as the users need them.

To format a diskette:

1. Turn the workstation off.
2. Insert the *standalone utilities* diskette (which came with the workstation) into the diskette drive.
3. Set configuration switch 5 down and switch 6 up. This selects the diskette drive as the boot device.
4. Set configuration switch 4 down. This lets you select the file you want to boot from the diskette.
5. Turn the workstation on.
6. The workstation prompts you with the following prompt:

   ```
   ))))
   ```

7. Enter:

   ```
   df(0,0)/saformat
   ```

   This loads the **saformat** program from the standalone utilities diskette, and provides you with the menu shown in Example 5–2.

```
        Drive Options

    1) Quit
    2) Winchester disk
    3) Flexible diskette

Select by entering a number from 1 to 3:
```

Example 5–2. Saformat top-level menu.

switch up when finished

8. Select item 3 from this menu. Once you have selected *Flexible diskette*, these messages appear:

   ```
   Put the diskette to be formatted into the drive.

   Press RETURN to continue
   ```

9. Remove the standalone utilities diskette from the diskette drive and return it to its protective jacket.
10. Insert the diskette you want to format. Remember whatever data was on the diskette is lost when the diskette is formatted.
11. Press ⟨RETURN⟩. The menu shown in Example 5–3 displays:

```
Defaulting to 48TPI Format

      Flexible Diskette Format Command Menu

   1) Return to previous menu
   2) Quit the formatting program
   3) Select alternate disk drive

   4) Set formatting information

   5) Sweep the disk surface for defects
   6) Format the disk with the given information

Select by entering a number from 1 to 6:
```

Example 5–3. Flexible Diskette Format Command Menu.

12. Select item 6 to use the default formatting values (if you want to check the diskette for defects first, select item 5; then, when the sweep is complete, select item 6). Once you have selected this menu item, numbers are printed on your screen as cylinders are formatted.

    When the formatting is done, the Flexible Diskette Format Command menu reappears on your screen. If you want to format more diskettes, go back to Step 10. Otherwise select the menu item to quit and press ⟨RETURN⟩.

Formatting a 61TC01/4944 Optional Hard Disk

CAUTION

Reformatting a hard disk should only be done by experienced system administrators. Reformatting a hard disk erases any data you have on that disk. Before you reformat the 61TC01/4944 optional hard disk, you should back up any data contained on the hard disk to the streaming cartridge tape. Also, the format program asks you many technical questions, such as the location of bad blocks on the disk. If you do not have access to this information, you cannot reformat the hard disk.

Reformatting the 61TC01/4944 optional hard disk takes about 45 minutes. The major steps to reformat the hard disk are:

1. Back up any data on the hard disk to the streaming cartridge tape. To back up data to the streaming cartridge tape use the *sysadmin* interface (see section 4). Specify the 61TC01/4944 optional hard disk as the source device, and the 61TC01/4944 streaming tape as the destination device.

2. Format the 61TC01/4944 optional hard disk by using the program **/etc/scsifmt**. This is an interactive menu-driven formatting program for the 61TC01/4944 hard disk.

   You use this program to specify which 61TC01/4944 optional hard disk by typing:

   `/etc/scsifmt` *device*

   For example, to format the entire 61TC01/4944 optional hard disk that is device number six and connected to the SCSI board in slot six, you would type:

   ```
   /etc/scsifmt /dev/rds66p
   ```

   The program asks you for information that it needs to format the hard disk, such as the bad block list. Some of the required information is rather technical. Casual uses of the workstation should not attempt to format the 61TC01/4944 optional hard disk.

   See also *makedev(5), makedev(8), sysconf(9).*

3. Run **/etc/newfs** (which constructs a new filesystem) on the hard disk. For more information on *newfs*, see the *newfs(8)* manual page.
4. Restore data to the hard disk from the back up media using the *sysadmin* interface (see Section 4).
5. Run **fsck** on the hard disk to clean up the file system. Also, add an entry to your */etc/fstab* file to periodically run **fsck** on your hard disk.

Setting the Time and Date

It is important for the workstation's clock to be properly set, because often users assign certain times for tasks to occur (**at** command). Also, you can set up system backups or other system tasks to occur automatically during periods of low system use. If the clock is incorrect, these jobs can start running at unexpected times, using system resources needed elsewhere.

The workstation clock should only need setting once, when you bring up the system for the first time. However, you should reset the clock if the workstation is moved to a new time zone, or if for some reason the clock stopped (such as, the computer board was changed).

The workstation's start–up procedure prompts you to check and correct the clock when more than 24 hours has passed since the workstation last had power. It does this by asking you to check the date.

To set the date and the clock, use the **date** command. The syntax for this command is:

date -z *timezone [yy]mmddhhmm[.ss]*

Where:

-z *timezone* is the time zone you are in. Table 5-4 shows valid time zone specifiers and the zones they represent.

yy is a two-digit field representing *year*. For example, 84 would indicate 1984. If you don't include this field, the last year entered is assumed.

mm is a two-digit field representing *month*. For example, 03 would indicate March.

dd is a two-digit field representing *day*. For example, 12 indicates the twelfth day of the month.

hh is a two-digit field representing *hours*. The 24-hour clock is used.

mm is a two-digit field representing *minutes*.

.ss is a two digit field preceded by a period (.) representing *seconds*. If you omit this field, the clock starts counting at .00 seconds of the minute you set with the second *mm*.

If you are doing system reconfiguration, you can also set the time using **sysconf**. See section 9 for details on setting the time using this method.

Table 5-4
TIME ZONE SPECIFIERS

Specifier (Standard Time)	**Specifier** (Daylight Time)	**Zone**
EET	EET	Eastern European
MET	MET	Middle European
WET	WET	Western European
AST	ADT	Atlantic
EST	EDT	Eastern
CST	CDT	Central
MST	MDT	Mountain
PST	PDT	Pacific
AEST	AEST	Australian: Eastern
ACST	ACST	Australian: Central
AWST	AWST	Australian: Western

System Shutdown

You can shut down your workstation in a variety of ways. You can shut the system down directly from multiuser (normal operational) mode, you can bring the system to single user mode and perform backups, then shut down from single user mode, or you can bring the system to single user mode, perform whatever operations you need to, then bring the system back to multiuser and shut down from there.

The recommended method of shutting down the system is to be in multiuser mode when you press the start/stop switch, whether you go to single user mode first to perform system tasks or not. This lets you use the 6130's *soft shutdown* capability.

Soft Shutdown

If the system is in multiuser mode, you can use the workstation's soft shutdown capability.

Use the **wall** command to warn all users of impending shutdown. Or, if you just brought the system from single user mode to multiuser mode, then, when all users are off the system, press the start/stop switch. The following then occurs:

1. The green light on the start/stop switch begins to blink, and a message similar to the following appears on the console screen:

   `Automatic shutdown in progress . . .` *day mon dt hh:mm:ss TZ year*
   `Performing disk checks . . .`

   Where *day* is the day of the week, *mon* is the month, *dt* is the day of the month, *hh:mm:ss* is the time, *TZ* is the time zone, and *year* is the year. For example, this series could be:

   ```
   Fri Aug 10 08:28:11 PDT 1984
   ```

   During the Automatic Shutdown, before the disk checks, all processes running on the system, except those necessary to complete the shutdown procedure, are killed.

2. At this point, the workstation is automatically running **fsck** in *preen* mode. **Fsck** reports any errors it finds to the console, and if it finds any errors it cannot correct, it exits and shuts down the workstation without completing the disk checks. If this occurs, **fsck** runs again next time the system is booted.

   If no errors occur, or if errors occur that **fsck** can fix while in *preen* mode, an **fsck** completion line appears on screen. This line looks something like this:

   ```
   /dev/dw00a: 1032 files, 9551 used, 3560 free (168 frags, 424 blocks)
   ```

   For more information on **fsck**, see Section 7 of this manual.

3. Once **fsck** is finished, the green light on the start/stop switch begins to blink faster. It does this for about one minute, then stops blinking altogether. At this point, the workstation is halted.

Getting to Single User Mode

When the system is in single user mode, it is as though you are logged on as root. You have absolute power to change any program or remove any file in the system. You should only enter single user mode when you need to perform administrative tasks that cannot be done from multiuser (normal operating) mode. These tasks include adding devices, restoring a large number of files, running a file system check (**fsck**), and so on.

CAUTION

When you are in single user mode, you can destroy the system by changing or removing vital system files. You should only use single user mode when you need to perform system administration tasks that cannot be performed while in multiuser (normal operating) mode. Use a regular user account with the system in multiuser mode for your personal work, and the sysadmin account for system administration tasks whenever possible.

When you enter single user mode (discussed in the following headings), the system prompts you for a password. Enter the root password. The system also prompts you for a terminal type with the line:

TERM = (unknown)

Enter the proper terminal type and press ⟨RETURN⟩. The system prompt when you are in single user mode is #, the same as the root account prompt.

Any time you are in single user mode, you can enter multiuser (normal operating) mode by typing ⟨CTRL–D⟩. When you press ⟨CTRL–D⟩, the following message appears:

```
Do you really want to leave single user mode? [y,n](n)
```

Type **y** in response top this question to bring the system to multiuser mode. If you type **n** or ⟨RETURN⟩ here, the system remains in single user mode.

The shutdown Command

Use the **shutdown** command to bring the system from multiuser mode to single user mode. If you use the **shutdown** command, the shutdown is announced to all users at intervals until shutdown, and login is disabled five minutes before the shutdown is due to occur.

There are a number of useful options to the **shutdown** command:

shutdown *+number* Brings the system to single user mode in *number* minutes.

shutdown now Brings the system to single user mode immediately. A message is sent to all users that shutdown is imminent, but they have no time to react.

shutdown -k *+number* Announces shutdown in *number* minutes, and disables logins five minutes before shutdown is scheduled, but doesn't really bring the system to single user mode. This is useful if you want to get all users off the system but want the system to remain in multiuser mode.

More information on the **shutdown** command can be found in *shutdown(8)*.

The Configuration Switches

You can also get to single user mode by using the configuration switches on the workstation back panel (see Figure 5–1). Or, more specifically, if you set configuration switch 4 to the down position, and answer properly the questions and prompts the system uses the next time you bring up the system, the workstation comes up in single user mode.

To use this method of bringing the system up in single user mode:

1. Turn off the workstation.
2. Set switch 4 to the down position (see Figure 5–1 for the location of the configuration switches on the workstation back panel).
3. Turn on the workstation.
4. When you see the ›››› prompt, press ⟨RETURN⟩.
5. The system goes through a few of tests, then it may inform you that the system configuration has changed since the last time it was booted (that is, the settings of the configuration switches has changed since the last time the system was started). The system asks if you want to change the *config* file to reflect the new configuration. Enter **y** and ⟨RETURN⟩, or just ⟨RETURN⟩ in response to this question.

   After you have answered the configuration question, the system finishes the tests and asks for a password.

   If this is not the first time you are bringing up the system after you've changed the settings of the configuration switches, the system does not ask the question, but runs through all the tests and then asks for a password.
6. Enter the root password in response to the password request.
7. The system prompts for a terminal type with the message:

   TERM = (unknown).

   Enter the proper terminal type and press ⟨RETURN⟩. The system prompt when you are in single user mode is #, the same as the root account prompt.

Shutdown From Single User Mode

If you don't want the system to go through the soft shutdown procedure when you turn off the workstation, bring the system to single user mode with the **shutdown** command.

If you just ran **fsck** with the system in single user mode, do not bring the system back to multiuser mode before you shut the workstation down. If you do, any repairs you made to the file system may be undone.

Before you turn the workstation off, make sure that all data is written to the Winchester disk by typing:

sync
sync

Do not use the **sync** *command if you did a file system check and the* **fsck** *repaired errors in the file system, as any such repairs may be undone.*

When the system is in single user mode, press the start/stop switch. The following prints to the console screen:

halting

The green light on the start/stop switch goes out, and the system halts.

If you shut the system down this way, **fsck** in *preen* mode runs next time the workstation is turned on.

6

Concepts for System Procedures

INTRODUCTION

This section discusses background concepts that you might find useful as you administer the system. These concepts expand upon the procedures covered in the previous two sections so as to give you a greater understanding of the UTek system.

The subjects covered in this section include:

- Boot files
- Users and groups
- Devices
- Daemons
- Sendmail

BOOT FILES

This discussion covers the programs that the system runs automatically every time the workstation goes from single user mode to multiuser (normal operating) mode. The system makes this change:

- During the start-up procedure.
- When you explicitly move from single user mode to multiuser mode by typing <CTRL-D> while in single user mode.

The file that the system calls at this time is */etc/rc*. This file is a shell script that contains the commands that are automatically run at the transition. The tasks this file performs include:

- Checking for the presence of */fastboot* and running **fsck** if */fastboot* isn't there.
- Mounting all file systems.
- Moving any editor files from the */tmp* directory to a holding place where users can reach them.
- Clearing the */tmp* directory.

The *rc* file also starts the system daemons, and calls the files that start the network and MDQS daemons, */etc/rc.net* and */etc/rc.mdqs*.

The file *rc.net* runs the **netconfig** program if this is the first time the system is being started. Then, *rc.net* starts the network daemons. The file tests if the network file system (NFS), standard network utilities, both, or neither are active. Then rc.net starts the daemons that are appropriate for the network configuration of your workstation.

The */etc/rc* file only calls the */etc/rc.mdqs* file.

The */etc/rc.mdqs* file starts the mdqs daemon for the MultiDevice Queuing System.

The */etc/rc* also calls */etc/rc.local*. This file does not exist when the workstation comes from the factory. If you want any commands run every time the system goes from single user to multiuser mode, create the file */etc/rc.local* and put the commands there.

USERS AND GROUPS

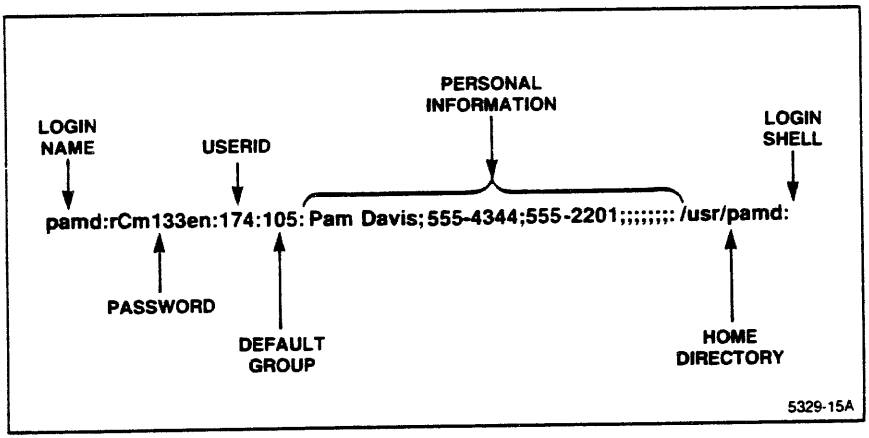
User and Group accounts are both ways of protecting users' files. These accounts and the file permissions associated with them allow users to protect their own files. Users can let everyone with system access read, modify, and execute a file, or they can restrict any or all of these privileges to themselves or members of their groups.

User Accounts

Users are the people who use the workstation. Each user should have a user account, and each user account needs a login name, a userid, a home directory and a login shell. The login name and the userid uniquely identify the user to the system. They are how the system knows who owns which files and processes.

One of the most common tasks that you will have to perform as a system administrator, especially early in the life of the workstation, is adding new users to the system. The sysadmin interface, discussed in Section 4, handles all of the account creation tasks for you — all you have to do is fill out the form on the screen.

When you add a user, the information you enter is added to the */etc/passwd* file, the appropriate home directories are created and their ownership properly set, the user is added to the appropriate group, and the default environment files are copied to the new home directories. Example 6-1 shows a sample one-line entry in the */etc/passwd* file.



Example 6-1. Sample /etc/passwd File Entry.

The fields in the Example 6-1 are:

Login name

Login names must be unique; that is, only one of each login name is allowed in the */etc/passwd* file (and therefore on the workstation). If the user is able to log in on more than one machine in a network, the login name should be unique in the network, too. You can ask what each user prefers as a login name, or you can arbitrarily assign names. Login names are often the first name of the user followed by the first initial of the user's last name, without a space in between. If there are conflicts, add another letter from the last name. The name can be up to 8 characters long.

Password

Users can choose their own passwords using the **passwd** command. If you are adding a new user, you can leave the password field empty while editing account information or assign a default password to all new users and instruct them to change it to one of their choice the first time they log in. The second method is preferable, so that you are not left with an unprotected account if the user neglects to enter a password. Good default passwords are *password*, *passwd*, or any other word you feel would be relatively easy to remember. The password you enter should be between 6 and 10 printable characters long.

Userid

The userid should also be unique to the machine and the network for security reasons. The userid should be an integer between 100 and 32767. You should reserve userids from 0 through 99 for the root account and other special accounts. If youare setting up a new user's account, the userid defaults to the next consecutive unused number. If a user is deleted from the system, the userid is not automatically reassigned (but you can reassign the number by entering it in this field).

If you are on a LAN, you might want to have a range of userids special to your system, so that userids on your workstation do not conflict with userids on other workstations on the network. If you are on a network, either see your network administrator for a range of userids, or, if there is no network administrator, get together with other workstation owners on your LAN and assign ranges.

Groupid

The default group for the user. The user can take advantage of any special group permissions on files owned by other members of this group, and those members can also share the user's files if so permitted. The groupid should be an integer between 0 and 32767. See the discussion on ranges of groupids later in this section.

Personal information

This field contains information like the user's full name, office extension, office number, and possibly home address and phone number.This is the information that is reported when someone uses the **finger** command with the user's name as the argument (see *finger(1)*). Notice that the subfields in this field are separated by semicolons (;) instead of colons.

The subfields in the personal information field are variable. You can set them by editing the */usr/lib/chfn* file. There is room for 10 subfields in the personal information field.

Users can change the information in this field with the **chfn** command. See *chfn(1)* for details.

Home directory

This field should contain the directory into which the system puts the user at login. A traditional home directory is */usr/login_name*, where *login_name* is the new user's login name. If you want another directory to be the user's home directory, enter that directory in this field.

More than one user can have the same home directory.

Login shell

This field should contain the full pathname of the shell that the user enters upon logging in: **/bin/sh** or an empty field if the Bourne Shell is to be the login shell, and **/bin/csh** if the C-shell (available with the UTek/A software package) is to be the login shell. These two shells are the only shells available with this release of UTek. If you install or develop another shell and want to use it, enter the full pathname of the file where the shell is stored.

Range of Userids

If your workstation is connected to a LAN, assign userids from a range that belongs exclusively to your workstation. Assigning a unique range of userids to each machine on a LAN is necessary so that no two users on the network have the same userid. The security of files on the network file system (NFS) depends on unique userids. For more information about security, see the *Network File System Reference* manual.

When a user on your workstation shares a userid with a user on another workstation on the network, NFS treats them as the same user. This allows files to be shared without specific permission.

To determine the range of userids you should use, see your network administrator if there is one. Otherwise, get together with other workstation owners on your LAN and choose ranges for each workstation (possibly saving ranges for future workstations). Remember that you can assign userids from 100 to 32767.

When you add a user to your workstation who already has an account on another machine on your LAN, use the userid from that other machine for that user. This is so users can log into your machine over the network and access their files without a password. This also keeps your range of userids from being used up quite as quickly.

Section 3 of this manual discusses NFS and other networking considerations. For more information about security, see the *Network File System Reference* manual.

Default Environment and Files

When you add a user with the sysadmin interface, the default environment files are automatically copied to the user's account. The default environment sets things like the default editor, the directories that the system looks in for the commands the user types, the terminal setup for the display being used, and default mail information.

The default environment files are files that set the user's environment at login. You can tailor the files that are copied into a user's account when it is created by editing the files that reside in */usr/lib/skeletons*:

- profile
- login
- cshrc

These files are automatically copied into the following files in the new account, respectively:

- .profile
- .login
- .cshrc

The *.profile* file dictates the login user environment when the user is using the Bourne shell, and the *.login* and *.cshrc* files dictate the user environment when the user is using the C-shell.

Users can personally tailor these files after they have logged in. For details on such changes, see the *6130 System User's Guide*.

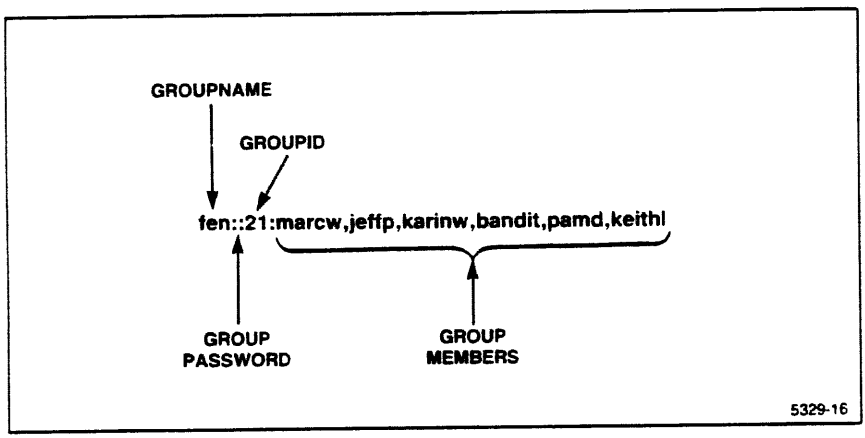
Group Accounts

You can assign users who need access to the same sets of files to *groups*. A group is a division used for file protection and permissions. You can assign users to a group, then they can all access files with group permissions set, and the files can still remain inaccessible by users not in the group.

Users can belong to groups. You set these groups when you add the users to the system (discussed in Section 4). Users can belong to one or many groups. A single user can belong to up to eight groups.

The list of group names is kept in */etc/group*. Each group name in the file is associated with the users who are members of the group. To add a user to the group without using the sysadmin interface, you must edit the */etc/group* file.

Example 6-2 shows a sample entry in the */etc/group* file.



Example 6-2. Sample Entry in the /etc/group File.

The fields in Example 6-2 are:

Group name	The name of the group. This name should be from 1 to 10 alphanumeric characters long, and start with a letter.
Password	This field is usually empty. You can assign a password to a group, but it is usually only done when a very high degree of security is needed.
Groupid	The groupid is used to determine whether a particular user is permitted to access a particular file according to the group permissions. The number should be between 0 and 32767.
Members of group	The login names of all the people in the group, separated by commas.

Range of Groupids

If your workstation is connected to a LAN, assign groupids from a range that belongs exclusively to your workstation. Assigning a unique range of groupids to each machine on a LAN is necessary so that the members of one group on a network file system cannot access the group files of another group. When a group on your workstation shares a groupid with a group on another workstation on the network, the distributed file system treats them as the same group. This allows files to be shared without specific permission.

To determine the range of groupids you should use, see your network administrator if there is one. Otherwise, get together with other workstation owners on your LAN and choose ranges for each workstation (possibly saving ranges for future workstations). Remember that you can assign groupids from 20 to 32767.

When you add a group to your workstation who already has an account on another machine on your LAN, use the groupid from that other machine for that group. This is so members of these groups can access the group files on your machine over the network without a password. This also keeps your range of groupids from being used up quite as quickly.

Section 3 of this manual discusses the network file system and other networking considerations. For more information about security, see the *Network File System Reference* manual.

UTEK DEVICES

There are two definitions for the word *device* for the 6130 workstation:

1. A hardware unit such as a disk drive, or interface board that is originally part of or an enhancement to the 6130 workstation.
2. A file in the */dev* directory that represents such a hardware unit for purposes of software communication.

If this discussion means definition 1, it uses the term *device*. If it means definition 2, it uses the term *device file*.

Each device or pseudodevice (discussed later) must have a corresponding device file in the */dev* directory. Each device file accesses a table of *device drivers* that resides in the kernel, and the name of the device file indicates which device driver responds to user requests.

Device drivers are the part of the system that does the actual communicating with devices. The standard kernel table of device drivers includes drivers for all the on-board standard devices and the 61TC01 cartridge tape drive. You must use the system configuration package (refer to the procedure in Section 9) to provide support for other devices.

Standard Devices

The standard devices already have device files on the 6130. To make sure that all these files exist on your workstation now, list the contents of the */dev* directory by typing:

```
ls /dev
```

Some of these files refer to specific hardware devices:

- The standard Winchester disk drive: */dev/dw00a* through */dev/dw00p*. There is only one system Winchester disk, the different final letters of the devices represent possible *disk partitions* on a Winchester disk. A disk partition is a section of the disk. The file system that the system uses for normal operation (the root file system) resides on device */dev/dw00a*.
- The diskette drive: */dev/df*.
- The two standard RS-232-C ports: */dev/tty00* and */dev/tty01*.
- The standard GPIB: */dev/gpib0*.

The Winchester disk, diskette drive, and GPIB also have device files associated with them that users should not try to access unless specifically instructed. For the Winchester disk, */dev/rdw00a* through */dev/rdw00p* refer to the disk partitions in *raw* mode. Diagnostic programs such as **fsck** use these device files. For the diskette drive, */dev/rdf* is used by the diskette formatting program, for the **dump** and **restore** programs when you use the sysadmin interface to backup and restore the system, and by the **cpio** program when you use the sysadmin interface to install new software. (Whenever you use these programs outside the sysadmin interface, you should also use */dev/rdf*.) For the GPIB, */dev/gpid0* is used by GPIB configuration software.

Some of the device files refer to *pseudoterminals*. These are the device files for logging in over the network. There are no hardware devices associated with these device files, but they act as though they represent terminals. These device files are */dev/ttyp0* through */dev/ttyp2* and */dev/pty0* through */dev/pty2*. See *pty(5)* for more information on these pseudoterminals.

There are also device files that correspond directly to the physical and virtual memory, the swap space, and to kernel tables. If you access the devices they refer to, especially if you write to these devices, you can destroy the system. These device files are:

- */dev/mem*
- */dev/kmem*
- */dev/drum*
- */dev/cvt*

Some device files can refer to different devices at different times:

- */dev/console* refers to the device that is currently the console. You can determine whether the console is a terminal connected to RS-232-C port 0 or a terminal connected to RS-232-C port 1 by looking at configuration switches 2 and 3 on the back panel of the workstation (see Table 6-1).

Table 6-1
CONSOLE DEVICE SETTINGS

Console Device	Switch 2	Switch 3
undefined	up	up
9600 baud RS-232-C terminal (port 1)	up	down
1200 baud RS-232-C modem/terminal (port 0)	down	up
300 baud modem/terminal (port 0)	down	down

- */dev/tty*, when accessed by a process, refers to the terminal where the process originated. If neither the standard input nor standard output of a process is the terminal where the process started, and the process must communicate with the user, it writes to */dev/tty*, and the message appears at the proper terminal. This means that */dev/tty* is different for all users logged in at the same time. (For a discussion of *standard input* and *standard output*, see Chapter 4 of *Introducing the UNIX System*.)

Finally, there is */dev/null*. The device file */dev/null* corresponds to nothing. If you have a program that outputs information that you don't want to see, redirect the output to */dev/null*. An end-of-file comes out of */dev/null* when you use it as input, and anything you output to it is thrown away.

Optional Devices

Several device drivers for optional devices are already present in the standard kernel.

The optional devices (or, *enhancements*) for which there are currently drivers in the kernel are:

- Optional high speed GPIB devices.
- SCSI streaming cartridge tape.
- Dual RS-232-C interface.

You can use the **MAKEDEV** utility to create device files for these devices. If you name the device files properly, they access the correct driver for the device you are adding. Section 5 discusses the procedure for creating device files.

DAEMON PROCESSES

Daemon processes, also known as daemons, are programs that automatically take care of various procedures for the system so that users do not have to invoke detailed individual programs every time they want to use one of these procedures. For example, there are daemons for network operations, printing operations, and mail system operations.

Daemons are started automatically each time the system moves from single user mode to multiuser (normal operating) mode (that is, whenever the file */etc/rc* runs). This includes whenever when UTek boots. Once started, daemons run in the background. They periodically check the appropriate places for jobs that they must take care of. The time interval between checks is set in the code for the individual daemon. Daemons are reported to the console when they start.

Daemons should not stop running unless you specifically stop them, but sometimes they do. If they stop, the task that they do does not get done. You can become aware that something is wrong with the task a daemon is supposed to do when a user complains, or if you notice something wrong.

Use the **ps -ax** command from the root account to find out if the daemon is running. If the daemon is not in the list that **ps** reports, you can start the daemon by typing the pathname of the file where the daemon resides. For example, if you needed to start the *nameserver*, a network daemon, type:

```
/etc/nameserver
```

You can also use the **ps -ax** command to see if there are multiple copies of the same daemon running. If there are, use the **kill** command to destroy all but one of the daemon processes. It doesn't matter which copy you leave running.

Table 6-2 lists the original daemons that come with the standard UTek system.

Table 6-2
ORIGINAL DAEMON PROCESSES

Daemon Name	File	Type	Function
syslog	/etc/syslog	network	Record error messages from other daemons
nameserver	/etc/nameserver	network	Convert hostname to address
routed	/etc/routed	network	Maintains network routing tables
udpd	/etc/udpd	network	Implements several network services (see *udpd(8n)*)
tcpd	/etc/udpd	network	Master network server (see *tcpd(8n)*)
talkd	/etc/talkd	network	Talk program daemon
nfsd	/etc/nfsd	network	Network File System daemon
mdqsd	/etc/mdqsd	mdqs	Multidevice Queuing System daemon
update	/etc/update	standard	Calls the **sync** command to update the contents of the superblock
cron	/etc/cron	standard	Run appropriate commands in */usr/lib/crontab*

Restarting Daemons

Sometimes you might change the configuration of the system by adding a device, changing the hostname or internet address, or doing something else that alters the way that daemons should see the system. If you do these things, you must restart the daemons that are associated with the change you made. The *Type* column in Table 6-2 will help you decide which daemons, if any, you have to restart after you change the system configuration. There are two ways to restart daemons:

1. Kill the daemon with the **kill** command, then restart it by typing the pathname of the file where the daemon resides.
2. Use the **shutdown** command to bring the system to single user mode, then type <CTRL-D> to bring the system back to multiuser mode. This method restarts all daemons.

Writing New Daemons

A daemon is simply a program that runs continuously in the background and performs a system service without user intervention. If you create a program that has these properties, and you want it to be started and stopped with the standard system daemons, put the program's pathname in the file */etc/rc.local*. You may have to create the *rc.local* file.

The *rc.local* file is run whenever the system moves from single user mode to multiuser (normal operating) mode. If it calls your daemon program, the program is started automatically the same way as the standard system daemons.

A traditional place to keep daemon programs is the */etc* directory.

SENDMAIL

Sendmail is a UTek tool that acts as a unified "post office" to which all mail can be submitted. It then sorts mail and sends it on to destinations that are on the local system or on other nodes on the local area network, or beyond, if it can find a node to relay the message to another network.

Sendmail does not have a user interface. To send and receive messages via the workstation mail system, users use the MH mail system, discussed in the *UTek Tools* book. The MH mail system sends messages to **sendmail**, which sorts the messages and hands them to the appropriate set of protocols to reach the appropriate destination.

There are three sets of protocols that **sendmail** knows about that deliver messages

- The software that delivers messages between users on the same system
- The SMTP protocols that send messages over the local area network using the LAN interface.
- The UUCP protocols that send messages over serial communications lines or modems to other computers.

Sendmail is already set up how to talk to the intramachine delivery software. This sysadmin interface allows you to set up the configuration file that **sendmail** uses to send messages to other machines.

The sendmail.cf File

The configuration file that defines how **sendmail** behaves is */usr/lib/sendmail.cf*. This file must contain the proper information for you to send mail to other machines. The *sendmail.cf* file can contain:

- The *local domain*. This is the name for the group of computers on your LAN. A local domain must be defined for **sendmail** to properly operate.
- Instructions on how to handle messages sent to an unknown host. You can specify that such messages be sent to a relay host (which you then specify), or that such messages generate an error message.
- A list of the machines that **sendmail** can send to via SMTP.
- A list of the machines that **sendmail** can send to via UUCP (copied from the uucp configuration files elswhere on the system).
- A list of the domains that **sendmail** knows about.

The file */usr/lib/sendmail.fc* is the frozen configuration version of the *sendmail.cf* file. Each time you change the *sendmail.cf* file with the sysadmin interface menus, **sendmail** reads the file and saves a configured version as *sendmail.fc*. Then, when **sendmail** runs in the future, it reads the *sendmail.fc* file and doesn't have to interpret the *sendmail.cf* file each time. This speeds up use of **sendmail**. Do not tamper with the */usr/lib/sendmail.fc* file.

Domains

A *domain* is a set of computers that can communicate with each other using **sendmail**. There are local domains, which contain the computers on a LAN, and there are superdomains, which contain one or more local domains. Figure 6-1 shows one way domains can be set up.

A computer can communicate via **sendmail** with any other computer in its local domain, and with any computer in any domain that is listed in the */usr/lib/sendmail.cf* file.

For **sendmail** to work properly, there must be a local domain in the */user/lib/sendmail.cf* file.

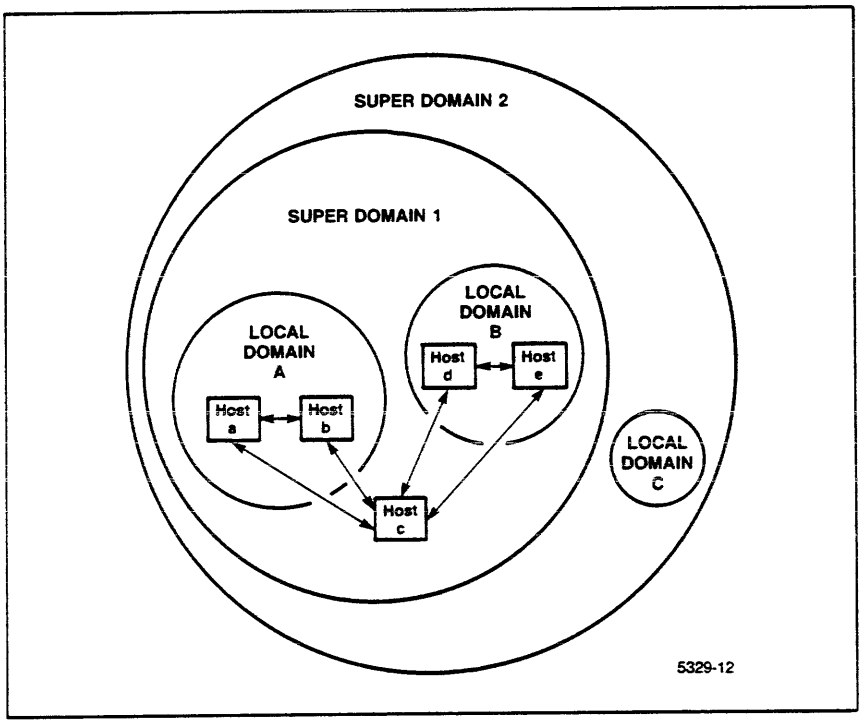


Figure 6-1. Sendmail Domains

7

System Health

Introduction

This section covers the tasks you have to perform to keep the system operating smoothly. These tasks are mostly maintenance and monitoring sorts of activities. They include:

- Preventive maintenance tasks, such as:
  - File system maintenance.
  - Making sure there's enough disk space for efficient operation.
  - Cleaning and maintaining the physical workstation.
- System messages reported to the console and what to do about them.
- Logging the system administration tasks you perform.
- Security.
- Dealing with the most common nonfatal problems (troubleshooting).

Read this section all the way through soon after you become the system administrator, so as to form a strategy for maintaining the system. Then you can refer to the appropriate parts of the section as you need to perform individual maintenance tasks.

Preventive Maintenance

The information in this section is designed to help you avoid system crashes and other catastrophes that might force you you to rebuild the system, since users' work is always lost when the system is rebuilt.

File System Maintenance

The utility called **fsck** (file system check) lets you repair the file system if it becomes corrupted. Corruption is defined here as:

- Blocks claimed by one or more inodes (file specifiers) or the free list.
- Blocks outside the file system claimed by an inode or the free list.
- Blocks not claimed by any file and also not on the free list.
- The wrong number of links to a file.
- A directory having an improper size format.
- An inode having a bad format.
- An inode whose number is out of range.
- A file associated with an unallocated inode.
- The number of blocks assigned to be inodes being is than the total number of blocks in the file system.
- The true number of free inodes is different from the number indicated in the superblock.
- The true number of free blocks is different from the number specified in the superblock.

If the file system does get corrupted, you can avoid having to restore the system in many cases by using **fsck**.

While **fsck** can't repair massive file system corruption, it can fix small to medium errors that deal with where a file goes in the file system.

The system runs **fsck** in *preen* mode every time the workstation is turned off while in multiuser mode. When **fsck** runs in preen mode, minor inconsistencies in the file system are automatically corrected. If the utility finds major corruption, it stops running, and notifies you with a message to the console device that it has stopped and the reason for stopping.

If the **fsck** at shutdown is not able to correct file system inconsistencies, the system tries **fsck** in preen mode again at the next start-up. If the program still isn't able to deal with the inconsistencies while in preen mode, it stops and the system comes up in single user mode. If this happens, you should run **fsck** in *interactive* mode. Interactive mode requires you to respond to prompts to correct more serious file system inconsistencies.

Constantly Running Systems

If you leave your workstation on constantly, run **fsck** in preen mode at least once a week. This fixes minor inconsistencies, and alerts you to potential problems before they become catastrophes. Run **fsck** in preen mode using this procedure:

1. Bring the system to single user mode. To do this, type:

   /etc/shutdown *+number*

   This **shutdown** command notifies any users that the system is going down in *number* minutes, then brings the system from multiuser to single user state in *number* minutes. See *shutdown(8)* in the *UTek Command Reference* manual for details on the **shutdown** command.

2. Type:

   fsck -p

If **fsck** finds something it cannot correct while in preen mode (a rare occurrence) it stops running and notifies you why it stopped. You should run **fsck** in interactive mode to correct the error.

Always run **fsck** when the workstation is in single user mode. This way you can be sure that nobody is writing to the file system as you are trying to repair it.

After the file system check is complete, type ⟨CTRL-D⟩ to return the system to multiuser (normal operating) mode.

Review of File System Structure

NOTE

This detailed information on file system structure and corruption is included to help you understand and repair the file system if **fsck** *in preen mode fails to repair the file system.*

For you to use **fsck** in interactive mode, you must understand the structure of a file system. Here is a quick review of the parts of a file system and their relation to each other.

Disk Partitions

File systems are mounted on *disk partitions*. A disk partition consists of some number of consecutive disk cylinders. There are 16 disk partitions on the Winchester disk. Each disk partition has a predefined purpose. Each partition corresponds to a device file in the */dev* directory. Table 7–1 lists the disk partitions and their purposes. As you can see, some partitions are included in other partitions.

Table 7–1
DISK PARTITIONS

Partition	Purpose
0	Root file system (/). Also part of partition 11
1	Swap space. Also part of partition 11
2	Reserved for future use
3	Reserved for future use
4	Reserved for future use
5	Reserved for future use
6	Reserved for future use
7	Reserved for future use
8	Reserved for future use
9	Reserved for future use
10	Reserved for future use
11	Data area (contains partitions 0 and 1)
12	Diagnostic test area
13	List of disk defects
14	Maintenance partition (boot area)
15	Entire disk

Boot Space (Partition 14)

The first few cylinders on a disk contain the bootstrap program (the disgnostics operating system), as well as a list of the bad blocks on the disk and other information that physically describes the disk.

Superblock

The next important block of a file system is called the **superblock**. It includes (but is not limited to) the following information about the file system. This block is in partition 0.

- The name of the file system
- The address of the superblock in the file system
- The size (in blocks) of the file system
- The number of data blocks in the file system
- The number of cylinder groups in the file system
- The size of blocks (in bytes)
- The size of block fragments (in bytes)
- The number of fragments in a block

The superblock is actually 8 kbytes long, longer than one 512–byte block, since the information it holds won't fit in a single block.

Cylinder Groups

A cylinder group is composed of one or more consecutive cylinders on a disk. A cylinder group is different than a disk partition in that cylinder groups are part of file system structure, while disk partitions are part of the structure of the disk itself. The bookkeeping information for each cylinder group includes:

- The number of cylinders in the cylinder group
- The number of inodes in the cylinder group
- The number of data blocks in the cylinder group
- The number of available *fragments* in the cylinder group
- A map of where used inodes are
- A *free block map* describing available blocks in the cylinder group
- A copy of the file system's superblock

A fixed number of inodes is allocated for each cylinder group when the file system is created. The current policy is to allocate one inode for each 2048 bytes of disk space; this should be far more inodes than will ever be needed.

The cylinder group bookkeeping information begins at a floating offset from the beginning of the cylinder group. If this information were at the beginning of each cylinder group, it would all be on the top platter. A single hardware failure that destroyed the top platter could cause the loss of all copies of the superblock. The offset for the information of the $n+1$st cylinder group is about one track further from the beginning of the cylinder group than that for the nth cylinder group. This way, the information spirals down into the disk; any single track, cylinder, or platter can be lost without losing all copies of the superblock. The space between the beginning of the cylinder group and the beginning of the cylinder group information stores data.

Superblock Copies

As mentioned earlier, each cylinder group includes a copy of the superblock. This way, if the original copy of the superblock is damaged beyond repair, the information stored in the superblock is not lost. You can specify the location (by block) of a copy of the superblock **fsck** should use with the **-b** option. There is always a copy of the superblock in block 32.

Fragments

To avoid waste in storing small files, the file system space allocator divides a single file system block into one or more *fragments*. Each file system block can be optionally broken into 2, 4, or 8 addressable fragments at the time the file system is created. The lower bound on the size of these fragments is limited by disk sector size; 512 bytes is the lower bound. The block map associated with each cylinder group records the space availability at the fragment level. Consecutive fragments that are part of the same block are examined to determine block availability. If an entire block is needed, available fragments of adjoining blocks cannot be combined to make up a block.

Free Block Map

Each cylinder group contains a map of the free blocks in the cylinder group. This map has information about all blocks in the cylinder group, and indicates which ones are used and which are free. The map records free fragments as well as free blocks.

Inodes

An inode is a file specifier. That is, each directory or file has one inode associated with it, and the information the inode contains is file–specific. Inodes include:

- The type of file the inode represents — data, directory, or special
- The number of links to the file or directory
- The owner's userid
- The owner's groupid
- The size of the file (in bytes)
- The last time the file was accessed
- The last time the file was modified
- The last time the inode was modified
- The addresses of the data blocks of the file

The number of inodes in a file system is specified in the superblock for that file system. There may be more than one inode in each block. The number of blocks that contain inodes depends on the number of blocks in the entire file system.

Data Blocks

Data blocks contain the actual data for directories and files.

Indirect Blocks

Indirect blocks contain pointers to data blocks or to other indirect blocks. A file contains indirect blocks if it becomes larger than a certain size. Each one of the 128 entries in an indirect block is a data block number.

There are three types of indirect blocks: single–indirect, double–indirect and triple–indirect. A single–indirect block contains a list of some of the data block numbers claimed by an inode. A double–indirect block contains a list of single–indirect block numbers. A triple–indirect block contains a list of double–indirect block numbers.

Before You Use Fsck

Before you run the **fsck** program on a file system you know is damaged, you should have a good archive copy of the file system you are repairing. However, don't try and take a backup of the damaged file system, as this may damage it even more.

Read through this section on **fsck** before you try to repair a file system for the first time.

Detection and Correction of Corruption

When **fsck** discovers an inconsistency, it reports the inconsistency so that you can choose a corrective action. The corrective actions described here can be performed interactively.

Superblock

One of the most common corrupted items is the superblock. Every change to the file system modifies the superblock. The superblock and its associated parts are most often corrupted when the workstation is halted without a **sync** command.

The superblock is checked for inconsistencies involving file system size, number of inodes, free block count, and the free inode count.

File System and Inode List Size

The file system size must be larger than the sum of the number of blocks in the superblock and the number of blocks used by the list of inodes. The file system and inode list sizes are critical pieces of information to the **fsck** program. While there is no way to actually check these sizes, **fsck** can check for their being within reasonable bounds. All other checks of the file system depend on the correctness of these sizes.

Free Blocks and Inodes

Fsck checks the free block map in each cylinder group. It ensures that all the blocks marked as free in the cylinder group block maps are truly free, that is, not claimed by any files. When all the blocks have been initially accounted for, **fsck** checks that the number of free blocks plus the number of blocks claimed by the inodes equals the total number of blocks in the file system.

If anything is wrong with the free block maps, **fsck** rebuilds them, based on the list it has computed of allocated blocks.

The superblock contains the total of free blocks within the file system. **Fsck** compares this total to the number of free blocks it found within the file system. If the two totals do not agree, then **fsck** replaces the incorrect total in the superblock with the actual free block count.

The superblock also contains the total of free inodes within the file system. **Fsck** compares this total to the number of free inodes it found within the file system. If the two totals do not agree, then **fsck** replaces the incorrect total in the superblock with the actual free inode count.

Inodes

An individual inode is not as likely to be corrupted as the superblock. However, because of the great number of active inodes, the inode list can become corrupted almost as often as the superblock.

Fsck checks the list of inodes sequentially, starting with inode 2 and going to the last inode in the file system. Each inode is checked for inconsistencies of format and type, link count, duplicate blocks, bad blocks, and inode size.

Format and Type. Each inode contains a *mode word*. This mode word describes the type and state of the inode. Inodes may be one of six types: regular inode, directory inode, symbolic link inode, special block inode, special character inode, or socket inode. If an inode is not one of these types, then the inode has an illegal type.

Inodes can be in one of three states: unallocated, allocated, and neither unallocated nor allocated. This last state indicates an incorrectly formatted inode. An inode can get in this state if bad data is written into the inode list through, for example, a hardware failure. The only possible corrective action is for **fsck** to clear the inode.

Link Count. Contained in each inode is the total of directory entries linked to the inode. **Fsck** verifies the link count of each inode by traversing down the total directory structure, starting from the root directory of the file system, calculating an actual link count for each inode.

If the stored link count is nonzero and the actual link count is zero, it means that no directory entry appears for the inode. If this occurs, **fsck** may link the disconnected file to the *lost+found* directory.

If the stored and actual link counts are nonzero and unequal, a directory entry may have been added or removed without the inode being updated. If this occurs, **fsck** may replace the stored link count by the actual link count.

Duplicate Blocks. Contained in each inode is a list or pointers to lists (indirect blocks) of all the blocks claimed by the inode.

Fsck compares the number of each block claimed by an inode to a list of already–allocated blocks. If a block number is already claimed by another inode, the block number is added to a list of duplicate blocks. Otherwise, the list of allocated blocks is updated to include the block number. If there are any duplicate blocks, **fsck** makes a partial second pass of the inode list to find the inode of the duplicated block. Without examining the files associated with these inodes for correct content, there is not enough information available to decide which inode is corrupted and should be cleared. Most times, the inode with the earliest modify time is incorrect, and should be cleared.

This duplicate block condition can occur by using a file system with blocks claimed by both the free list and by other parts of the file system. A large number of duplicate blocks in an inode may be due to an indirect block not being written to the file system. **Fsck** prompts you to clear both inodes.

Bad Blocks. Each inode contains a list or pointer to lists (indirect blocks) of all the blocks claimed by the inode.

Fsck checks each block number claimed by an inode for a value lower than that of the first data block, or greater than the last block in the file system. If the block number is outside this range, the block number is a bad block number, and the block is counted as a *bad block* for **fsck** purposes. (Do not confuse an **fsck** bad block with a bad disk block; they are not related.)

A large number of bad blocks in an inode may be due to an indirect block not being written to the file system.

Fsck prompts you to clear the inode.

Size Checks. Each inode contains a count of the number of data blocks that it contains. The number of actual data blocks is the sum of the allocated data blocks and the indirect blocks. **Fsck** computes the actual number of data blocks and compares that block count against the actual number of blocks the inode claims. If an inode contains an incorrect count, **fsck** prompts you to fix it.

Each inode contains a 32-bit size field. The size is the number of data bytes in the file associated with the inode. The consistency of the byte size field is roughly checked by computing from the size field the maximum number of blocks that should be associated with the inode, and comparing that expected block count against the actual number of blocks the inode claims.

Indirect Blocks

Indirect blocks are owned by an inode. Therefore, inconsistencies in indirect blocks directly affect the inodes that own them. **Fsck** checks if the indirect blocks are already claimed by another inode (see the Duplicate Blocks section discussed previously) and if block numbers are outside the range of the file system (see the Bad Blocks section discussed previously).

Data Blocks

The two types of data blocks are *regular data blocks* and *directory data blocks*. Regular data blocks contain the information stored in a file. Directory data blocks contain directory entries. **Fsck** does not check the validity of the contents of regular data blocks.

Each directory data block is checked for inconsistencies involving directory inode numbers pointing to unallocated inodes, directory inode numbers greater than the number of inodes in the file system, incorrect directory inode numbers for "." and ".." directories, and directories that are disconnected from the file system.

If a directory entry inode number points to an unallocated inode, then **fsck** may remove that directory entry. This condition probably occurred because the data blocks containing the directory entries were modified and written to the file system while the inode was not yet written out.

If a directory entry inode number is pointing beyond the end of the inode list, **fsck** may remove that directory entry. This condition occurs if bad data is written into a directory data block.

The directory inode number entry for the "." directory should be the first entry in the directory data block. Its value should be equal to the inode number for the directory data block.

The directory inode number entry for the ".." directory should be the second entry in the directory data block. Its value should be equal to the inode number for the parent of the directory entry (or the inode number of the directory data block if the directory is the root directory */*).

If the directory inode numbers are incorrect, **fsck** may replace them by the correct values.

Fsck checks the general connectivity of the file system. If directories are found not to be linked into the file system, **fsck** links the directory back into the file system under the *lost+found* directory. This condition can be caused by inodes being written to the file system with the corresponding directory data blocks not being written to the file system.

Using Fsck

The **fsck** utility makes a number of passes through the file system.

When you are running **fsck** interactively (that is, when you issue the **fsck** command without the **-p** option), to run the utility twice if **fsck** reveals a damaged file system. Run it the first time to give you some idea of what exactly is wrong, and the second time to correct the problem. If the automatic **fsck** performed when bringing up the system finds a problem, consider that your first run, and answer *no* to any questions the utility asks you (except those that ask you if you want to continue checking the file system, answer **y** to these). Then, run **fsck** a second time to repair the file system.

Always run **fsck** *from the workstation's single user mode.* If you run **fsck** while the workstation is in multiuser (normal operating) mode, some file system errors may remain uncorrected.

The Phases of Fsck

Fsck goes through a number of phases. These are:

Initialization	Before a file system check can be performed, certain tables must be set up and certain files opened. This phase also contains messages resulting from command line options, memory requests, opening of files, status of files, file system size checks, and creation of the scratch file.
Phase 1	*Check Blocks and Sizes.* This phase concerns itself with the inode list. Error conditions arise from incorrect inode types, problems in setting up the zero link count table, bad and duplicate blocks, incorrect inode size, and incorrect inode format.
Phase 1b	*Rescan for More Duplicates.* When a duplicate block is found in the file system, the file system is rescanned to find the inode that previously claimed that block.

Phase 2	*Check Pathnames.* Lets you remove bad and duplicate data blocks from files and directories. Also, lets you remove directory entries for inodes whose information is not salvageable. This phase deals with error conditions resulting from root inode mode and status, directory inode pointers out of range, and directory entries pointing to bad inodes.
Phase 3	*Check Connectivity.* **Fsck** checks that all directories are reachable by trying to connect to each one. This phase deals with error conditions resulting from unreferenced directories, and missing or full *lost+found* directories.
Phase 4	*Check Reference Counts.* Clears out any bad inodes found in Phase 1. Checks that all files are reachable by trying to each one. Checks and fixes the link count in each file header. This phase deals with error conditions resulting from unreferenced files, missing or full *lost+found* directory, incorrect link counts for files, directories, symbolic links, or special files, unreferenced files, symbolic links, and directories, bad and duplicate blocks in files, symbolic links, and directories, and incorrect total free-inode counts.
Phase 5	*Check Cylinder Groups.* Lists error conditions resulting from blocks in the free block map, free blocks missing from the free block maps, and incorrect total free block count.
Phase 6	*Salvage Cylinder Groups.* Rebuilds the free block maps. This phase doesn't generate any messages.
Cleanup	Once a file system has been checked, a few cleanup functions are performed. This section lists advisory messages about the file system and the modify status of the file system.

Appendix A of this manual contains a detailed list of the **fsck** messages.

After the Fsck

iAfter the **fsck** has finished, if any changes were made to the file system, restart the workstation using the start/stop switch. Do not bring the workstation to multiuser (normal operating) mode before restarting it.

Disk Space

It is very important that there be enough free disk space on the Winchester disk. If the disk is too full, the system slows down and may stop.

You can determine how full the Winchester disk is with the **df** command. Enter the command:

df

This tells you:

- The names of the file systems on the Winchester disk.
- The number of kbytes total in each file system.
- The number of kbytes currently in use in each file system.
- The number of kbytes left to use in each file system.
- The percentage of each file system currently in use.
- The directory that the file system is mounted on.

The 6130 workstation has only one file system when you get it from the factory. This file system is mounted on the root directory, /. Example 7–1 shows a sample response to the **df** command. See *df(1)* for more information on the **df** command.

```
Filesystem    kbytes   used   avail  capacity  Mounted on
/dev/dw00a     13111  10279    1520     87%    /
```

Example 7–1. Sample Response For df Command.

The capacity as reported by the **df** command should be below 90%. To keep this capacity from getting too high, remove unnecessary files on a regular basis. Files that you can remove without causing trouble for users are:

- Files in */usr/spool/rwho*. Every time the network daemons contact a new node, a file for that node is created in this directory. If you are on a network with a lot of nodes constantly being added or deleted, you may want to periodically empty this directory.
- Files that start with a comma (,). Remove such files older than a day or two.
- Files that start with a number sign (#). Remove such files older than a day or two.
- All *core* files. Remove such files older than a day or two.

There are other files that are constantly getting larger because the system is constantly writing to them. These files are:

- The file */usr/adm/messages*. This file contains a record of all system messages that are reported to the console.
- The file */usr/adm/wtmp*. This file contains a running record of who logs onto or off of the system.
- The file */usr/adm/shutdownlog*. This file contains messages that are generated each time you shut down the workstation.

You can keep these files to a manageable size by removing them on a regular basis, or by copying the contents to diskette on a regular basis, say, once a week. The system then creates the file over again, but it never contains more than one week's worth of data.

You may also want to remove a part of these files and save the most recent information in them. For example, if you wanted to save the last 100 messages written to */usr/adm/messages*, use the following series of commands:

```
tail -100 /usr/adm/messages >/tmp/file
cp /tmp/file /usr/adm/messages
rm /tmp/file
```

Where *file* is any file name (use the same file name in all these commands).

You can run commands like this manually on a regular basis, or you can put a series of commands like this into a shell script and put the shell script into */usr/lib/crontab* for the **cron** daemon to run regularly (see the discussion on **cron** and */usr/lib/crontab* later in this section).

The advantage to moving the information in one of these files to another file is that you then have a record of the information that goes back as far as you want it to.

For example, suppose you wanted to save two week's worth of the system messages that are automatically written to the file */usr/adm/messages*. Each week, move the contents of */usr/adm/messages* to */usr/adm/messages.old*, or some such file with the **mv** command. This overwrites the old *messages.old* file (from three weeks ago). The system recreates */usr/adm/messages* the first time after the move that it has something to add to the file. This method always gives you the current and the previous weeks' messages.

An example of this method is the shell script */usr/adm/newsyslog*, which is run nightly by the **cron** daemon (discussed next). This shell script moves the information from */usr/adm/syslog* to */usr/adm/syslog.0*, from */usr/adm/syslog.0* to */usr/adm/syslog.1*, and so on through */usr/adm/syslog.7*. If you really need disk space, you may want to remove these *syslog.n*. However, don't remove *syslog*, since it contains the current day's messages.

You can keep more than two weeks' information by moving *messages.old* to, say, *messages.3* each week, giving you three weekly message files, and so on. You can choose any time interval to perform these moves.

You can also remove files in */usr/tmp*, but you should check with the files' owners first. Since any user can write to */usr/tmp*, any user may own a file in that directory.

The cron Daemon and crontab

Crontab is a table of commands that is read by the *cron* daemon once a minute. If a command in the table is scheduled to be executed that minute, it is.

The format of the commands in */usr/lib/crontab* is discussed in *cron(8)* in the *UTek Command Dictionary*.

Example 7–2 shows a sample */usr/lib/crontab* file. This *crontab* file:

- Appends the file */etc/dmesg* to */usr/adm/messages* every 10 minutes.
- Runs the program **/usr/lib/atrun** every 15 minutes.
- Runs the **calendar** program at 4:00 a.m. daily.
- Runs the program **/usr/adm/newsyslog** at 4:05 a.m. daily.
- Finds and removes all files in the directory */usr/preserve* that are more than seven days old at 4:15 a.m. daily.

You can set up various files to perform various system cleanup tasks, and then put these files in command lines in */usr/lib/crontab* to be automatically executed by the cron daemon. These files are called *shell scripts* and each one can contain a command or list of commands to perform the desired task. Tasks that you can write such shell scripts for include:

- Finding and removing files automatically, as done in Example 7–2.
- Moving the information in one file to another file to keep the first file to a manageable size, as done by the **newsyslog** shell script called in Example 7–2.
- Notifying you that it's time to take a system backup.
- Any other task that you want the system to perform automatically.

```
0,10,20,30,40,50 * * * * /etc/dmesg - >>/usr/adm/messages
0,15,30,45 * * * * /usr/lib/atrun
00 04 * * * calendar -
05 04 * * * sh /usr/adm/newsyslog
15 04 * * * find /usr/preserve -mtime +7 -a -exec rm -f {} ;
```

Example 7–2. Sample /usr/lib/crontab File.

Periodic System Backups

It is *very important* to take periodic backups of both user data and the entire system. This way, you can restore users files if they have been destroyed, and most importantly, you have a recent copy of the system should something happen and you need to restore the system.

You can use the sysadmin interface to take backups. The procedure for taking backups is discussed in Section 4. Here, strategies for scheduling different types of backups are discussed.

The syadmin interface uses the **dump** command to back up the system. For more information about **dump** and backup levels, see *dump(8)*.

When To Take Backups

Take a level 0 (full system) backup soon after you boot the system for the first time. Back the system up *after* you have installed any optional software packages and configured the system for any hardware enhancements. This spares you the need to reinstall any optional software should you ever need to restore from the backup. Keep the diskettes or cartridges in a safe place. This ensures that you have an uncorrupted version of the system in reserve.

You may want to take two total system backups at this time, and store the two sets of backup media in different places.

A total system backup can take as long as two hours for a 40 Mbyte Winchester disk. The more data to be backed up, the longer the backup takes.

Also, if you are backing up to diskette, be sure to have enough formatted diskettes before you begin a any backup. It takes about three formatted diskettes for every Mbyte of data. For example, if you have a 40 Mbyte Winchester disk, about 27 Mbytes are available for use, so you should plan to have about 81 formatted diskettes ready to use for a level 0 backup. When you start the backup, the **dump** command tells you the approximate number of diskettes that the backup needs.

Always use quality diskettes to protect your data. If you need diskettes, contact your Tektronix Field Office (the part number for a box of 10 diskettes is 119–1583–00). Information on formatting diskettes is in Section 5.

If the system users are creating or changing many files, or if having current backups is very important to you and the users, you might want to take an incremental backup daily or semiweekly. Otherwise, weekly incremental backups are probably sufficient.

These incremental backups should be Level 9, so that they record all changes made since the previous backup. Each time the number of diskettes needed for a level 9 backup gets larger than about five, take a level 5 backup to decrease the size of subsequent level 9 backups.

Besides the semiweekly (or weekly) incremental backups, you should take one total system (level 0) backup each month. You may want to have two copies of this total system backup to guard against media problems.

Verifying the Backup

The **dump** program writes the directory of the backup on to the backup media before it begins the actual backup. If you take backups on diskette, you should have two copies of this directory for each backup, in case the diskette(s) that contains the directory is damaged (with a cartridge tape, since the entire backup is on one tape, an extra copy of the directory on another tape is not much use).

To get this second copy of the directory:

1. Find out how many diskettes you need to copy to assure getting the entire directory by putting the first diskette of the backup into the drive and restoring from it (use the Restore capability in the sysadmin interface, see Section 4). Remove and insert subsequent diskettes as directed until you see the **restore**› prompt that indicates you are in interactive restore mode, then abort the restore (instructions for aborting an interactive restore are avalable in Section 4). Count the diskettes you have already restored (including the current one). This is how many diskettes you need an extra copy of. Usually, you do not need to copy more than one diskette.
2. When you know how many diskettes you need to get the whole directory, use the Backup capability in the sysadmin interface (see Section 4) to backup this many diskettes, then abort the backup. Instructions for aborting a backup are provided by the **dump** command as you use the sysadmin interface's Backup procedure.

Storing and Recording Backups

Label each backup diskette or cartridge as you take it out of the drive. The first diskette/cartridge is *volume 1*, the second is *volume 2*, and so on. Do this so that you can keep the volumes in order if you ever have to restore from them.

Keep your semiweekly and weekly backups for a month before reusing the diskettes, and keep your monthly full system backups for six months. If you can, store a copy of the previous month's total backups somewhere off-site. That way, if something happens at the workplace (fire, theft, etc.), you never lose more than a month of work.

System Messages

There are many system messages that can appear on the console screen. These are messages that the kernel sends to the console to note occurrences during system operation. Most of these messages are for information purposes only. They let you know what is happening on the system at an internal level. This manual does not list these or explain these information messages.

panic Messages

Occasionally the system notifies you, by sending a *panic* message to the console device, of the cause of a sudden halt immediately before it happens.

The purpose of panic messages is to give you some idea of the reason for a catastrophic system failure. There is not much time for the system to display the message before the crash, so the information you get can be rather cryptic.

A properly operating system should never generate a panic message. If your system sends a panic message to the console device, copy it down, and contact your Tektronix Service Representative immediately. You should also make carefully written notes on what the system was working on immediately before the panic message appeared and the system died.

Diskette Distribution

When you got your workstation, you also received a set of nine diskettes. These are:

- The *standalone utilities* diskette, containing utilities that can run without UTek. These include **saformat**, which formats the Winchester disk, and **sacopy**, which copies data between devices when UTek is not available.
- The three-volume *miniroot*, which, when copied the the Winchester disk, contains the minimum amount of information for the system to boot. However, you cannot boot the system with only the miniroot, you need to specify a kernel file.
- The *miniroot system* diskette, which contains a copy of the UTek kernel. Use this to boot the system if the kernel on the Winchester disk is not usable.
- The four *system configuration* diskettes, which allow you to configure the kernel by using the *sysconf* interface. You should configure the kernel when you first install the workstation. For more information, see Section 9 of this manual.

Keep these diskettes in a safe place. Guard them not only from harm, but also from theft, since your 6130 cannot be kept secure from someone who has copies of these diskettes.

The use of these diskettes is discussed extensively in Sections 5, 8, and 9.

Logging Administration Tasks

Whenever you perform most system administration tasks, you should record at least the day and time you did them, and what you did. You can do this by editing a file that you create specifically for this purpose, or by sending yourself mail.

If possible, you should also keep a paper record of tasks you perform. This way, you can still access the information even if the system is down.

Table 7–2 shows the tasks you should record, and what information you should record about them.

Table 7–2
SYSTEM ADMINISTRATION TASKS

Task	Information
Bringing the system up	Date, time, device you got kernel from (if not Winchester disk), and whether there were errors or not.
Bringing the system down	Date, time, and reason.
Bringing the system to single user mode	Date, time, and reason.
Backing up the system	Date, time, **dump** level, list of files backed–up, media dumped to, and number of volumes.
Restoring files	Date, time, list of files restored, media and volume number restored from.

Security

A major concern of many system owners today is security. Is their system secure? Can it withstand attempts at illegal entry? Can private information be protected while remaining accessible to users who need it? Can the system itself be protected so that its functionality is not impaired?

To get full use from this section on security, you should understand the discussion in Chapter 3 of *Introducing the UNIX System* on file and directory ownership, protection, permissions, and the **chmod** command.

The main security strategy of UTek is to keep intruders from logging on. An equally important security concern is to keep users from obtaining superuser privileges and corrupting the system or others' files.

Once someone is inside the system, security depends mostly on the protection schemes assigned by the superuser and by individual users. This is why passwords are so important. If a single user does not assign a password, if users don't keep their passwords secret, or if users use obvious, easy to guess words as their passwords, it becomes very difficult to protect the system and the users' data.

Even more dangerous can be if an unauthorized person obtains superuser privileges. If this occurs, there is really no way to protect the system, due to the power that the superuser has. In fact, if someone manages to get superuser privileges once, that person can leave all sorts of trapdoors in the system, so that even if you close the original hole, the intruder can get in through the trapdoors.

How UTek Keeps Out Intruders

UTek uses a one-way encryption algorithm on the password that a user types in. It then compares the result with the encrypted password in the user's entry in the */etc/passwd* file. If the two are the same, the user is logged on. Otherwise, the user cannot log onto the system. Note that only the encrypted version of the password is kept in */etc/passwd*.

Protecting Superuser Privileges

There are a number of methods you can use to keep the unauthorized from obtaining superuser privileges. These are:

- Guarding the passwords for special accounts.
- Changing the passwords for special accounts regularly.
- Keeping an eye on *set user ID* programs.
- Assigning a unique userid to each user on the network, if your workstation is connected to one.

Passwords to Special Accounts

You should always keep the passwords to accounts with root privileges secret. If you must share them, tell them only to people you can trust with the system. Also, to keep these passwords from becoming general knowledge, change them regularly. Once a month is not too often to change the passwords of the accounts that allow superuser privileges. These accounts are:

- root
- sysadmin

NOTE

While keeping superuser passwords secret is important, at least two people should have the root password for any given system. If you are the only one with the root password, difficulties could arise if you are not available when the system needs the attention of a superuser.

Set User ID Programs

A concept that is important to protecting superuser privileges is that of real and effective userids and groupids, and of programs that use the setuid (SUID) and setgid (SGID) system calls.

Real and Effective IDs

UTek programs are assigned ID numbers that indicate which files and directories the program can access. A real ID is the same as the userid of the person who started the program running. An effective ID is the ID that actually sets the permissions that a program is allowed.

Effective IDs can be a number of things, depending on which user owns the program being run.

- The default effective ID is the same as the real ID; that is, the same as the userid of the user who started the program.
- If the program is specifically marked, the effective ID can be the same as the userid of the user who owns the program.

Groupids are treated the same way as userids, with both real and effective groupids.

The way to tell if a program is a set-user-ID (SUID) or set-group-ID (SGID) program is to look at the permissions of the file holding the executable version of the program. There is an *s* where the *x* would be in the three letters indicating owner permissions if the program is an SUID program, and in the three letters indicating group permissions if the program is an SGID program.

For example, the permissions for a nonSUID program might be:

```
-rwxr-xr-x
```

While those for an SUID program with the same permission mode would be:

```
-rwsr-xr-x
```

And those for an SGID program with the same permission mode would be

```
-rwxr-sr-x
```

Why Worry about Effective IDs?

The concern about effective IDs is that there are quite a few system programs owned by the root that set the effective ID to root, giving the user who invoked the program superuser privileges while the program is running. These programs are usually well protected enough that they don't provide holes for an intruder, but someone who obtains the root (or another user's) password can create SUID programs as trap doors that allow future access to otherwise forbidden areas of the system. Or, a clever intruder could find a bug that has been previously missed, allowing that user to retain superuser privileges after the program has finished running.

Soon after you get your system, use the following command to determine which programs legitimately set the userid to root (SUID root programs):

find / -type f -user 0 -perm 4000 -print \;>*filename* **&**

This sends the list of SUID root programs to the file *filename*. The ampersand (**&**) in the command line puts the command in background mode, since the command takes so long to complete. Save this file for future reference.

Then, periodically run this same **find** command into a different file and compare the two files to determine if there are any new SUID root programs. If there are, and you didn't create them or permit their creation, try running them from a nonroot account to see what they do. If the action of the program seems dangerous, or if you don't trust the owner of the file, you should take action to keep the intruder from doing more damage (remove the file, seek disciplinary action against the intruder, and so on).

If a user comes to you with tales of tampered files, you can use the **find** command to discover if there are any SUID programs that set userid to that user's userid. The command to use is:

find / -type f -perm 4000 -exec ll {} \; >*filename* **&**

This will find all the SUID programs on the system, and give a long directory listing for them. The ampersand (**&**) in the command line puts the command in background mode, since the command takes so long to complete. You can view the file *filename* with **more** or **cat** to find out if the someone has created a SUID program that makes the user's userid its effective ID, or use **grep** to find the user's login name in the file.

Clear all such programs with the the user having problems. If one or more programs can be identified as not valid, use your power as superuser to remove the program (with the **rm** command), and have the victim change passwords (with the **passwd** command).

Guarding Against Trojan Horses

Another thing you should watch out for is running users' programs while you are the superuser. It is possible for a user to write an SUID program such that the user can get the privileges of the person running the program. If you run this user's program while you are root, the user can then get superuser privileges. Therefore, it is a good idea to not be in the root account when you run users' programs. Run them from your personal account, instead, just in case.

You should also be careful to keep publically writable directories out of the $PATH variable in the root account's .profile and .cshrc files. This is the variable that defines where the system looks for the command you enter. If a directory of this sort is in that variable, someone can create a file with the same name as a commonly used command that does more than that command.

For example, someone could write a file that changed the password of the person running the command, then name the file *cat* and put it in a public directory. If this directory was in the $PATH variable for the root account, then when you run **cat** as root, you might run this false file, which would change the root password without your knowledge.

Possible System Problems (Troubleshooting)

Although problems in UTek system operation are not common, they can occur. This section deals with the the most common nonfatal problems that can happen: what causes them, how to track down the cause, and how to recover from the problem. The problems discussed in this section do not halt the system entirely, they just make the system difficult to work with. See Section 8, System Halts, for a discussion of fatal problems.

This section presents the problems in a trouble-shooting manner; that is, it allows you to look up the problem via the symptoms, and then presents possible causes and solutions for the symptoms.

This discussion cannot cover everything that can cause the system problems. If you cannot find the symptoms of your system's problem in this discussion, contact your Tektronix Service Representative.

Forgotten Password

If you forget a password for an account that is not the root account, you can use log in as root and change the password to one you know with the **passwd** command.

For example, if you forget the password to the *sysadmin* account, log in as root. Then:

1. Type:

   passwd sysadmin

2. The system responds:

   Changing password for sysadmin
   Enter new password:

   Enter the new password and press ⟨RETURN⟩. The password you enter is not displayed for security reasons.

3. The system then responds:

   Retype new password:

   Retype the password you just entered and press ⟨RETURN⟩. Again, the password you type is not displayed.

This password for the sysadmin account is now changed. From the root account, you can change any password in this manner. Just substitute the *sysadmin* in the above example for the login name on the account whose password you want to change.

Forgotten User Password

Users may come to you and tell you that they forgot their passwords. Use the previous procedure to change the password on their accounts, and then tell them what you changed it to. Instruct them to change it to something new immediately. If you assign them a password like "forgot", they will likely change it more quickly than not.

Forgotten Root Password

If you forget the root password, and if there is no-one else around who knows it, the procedure for getting into the root account is much more difficult than the one for restoring passwords to regular accounts.

To recover from a forgotten root password, you must use the standalone utilities diskette, the three miniroot diskettes, and the miniroot system diskette from the diskette distribution (discussed earlier in this section). You must load the miniroot into the workstation, boot the workstation from the miniroot system diskette, mount the regular file system onto the miniroot, and edit the */etc/passwd* file to remove the password field of the root entry.

Recovery

1. Turn off the workstation.
2. Set configuration switches 5 and 6 to specify diskette (down, up). Set configuration switch 4 to down. This lets you select a file on the diskette to boot from.

3. Insert the *standalone utilities* diskette into the diskette drive.
4. Turn on the workstation.
5. When the ›››› prompt appears, type

   df(0,0)sacopy

   Do not enter a space between the right parentheses **)** and **sacopy**.
6. The program prompts you to make sure that the first volume (diskette) of the miniroot is in the diskette drive. remove the standalone utilities diskette, insert the miniroot diskette, volume 1, and press ⟨RETURN⟩.
7. The program prompts you for the name of the device you want to copy *from*. Press ⟨RETURN⟩ here, as the device defaults to df(0,0), the diskette drive.
8. The program then prompts you for the device you want to copy *to*. Press ⟨RETURN⟩ here, as the device defaults to dw(1,0), the Winchester disk.
9. The program asks you to enter the block size and the number of records to copy. Press ⟨RETURN⟩ in answer to these questions; the block size defaults to 10240 and the number of records defaults to to 108.

   This begins loading the miniroot onto the Winchester disk's swap space. Dots are printed to the screen to indicate that the **sacopy** program is running.
10. When you see the message:

    ```
    Read n records from 1 volume(s)
    Insert next volume. Press RETURN to continue.
    ```

    remove volume 1 and insert the volume 2 miniroot diskette. Press ⟨RETURN⟩ to continue. The *n* in the message is the number of records copied so far from the miniroot diskette to the Winchester disk.
11. When you see the message in Step 7 again (except it informs you that it has read from 2 volumes), remove volume 2 and insert the volume 3 miniroot diskette. Press ⟨RETURN⟩ to continue.
12. When the copying of volume 3 is complete, the **sacopy** program asks you to insert the system diskette. Remove the volume 3 miniroot diskette, insert the miniroot system diskette, and press ⟨RETURN⟩.

When the miniroot system is installed, load the UTek kernel from the miniroot system diskette with the following procedure:

1. When the ›››› prompt appears, type:

   df(0,0)vmunix

   Do not enter a space between the right parentheses **)** and **vmunix**. If you just press ‹RETURN› without entering anything here, the file that the system chooses automatically defaults to *vmunix*.

2. When it asks you for a root device, type:

   dw00*

   This specifies the swap space, where the miniroot was loaded.

You should then see the root prompt, #. This indicates that UTek is running in single user mode, and that you are logged in as *root* on the miniroot.

Once you have the miniroot loaded, you must make the standard root file system on the Winchester disk accessible, so that you can reach the */etc/passwd* file.

1. Type:

   fsck -y /dev/rdw00a

   This checks the file system on the Winchester disk.

2. Type:

   mount /dev/dw00a /mnt

   This mounts the root file system from the Winchester disk onto the directory */mnt* on the miniroot.

3. Now type:

   cd /mnt/etc

   This moves you to the */etc* directory on the Winchester disk. (Remember you mounted the root file system to the */mnt* directory on the miniroot.)

4. Now, edit the *passwd* file. The first line in the file looks like:

   root:xxxxxxxxx:0:0:root;;;;;;;;;:/.:

   Remove the second field in that line so that the line looks like:

   root::0:0:root;;;;;;;;;:/.:

   This removes the root password.

5. Exit the editor, saving the change you made and making no other changes to the *passwd* file.

6. When you are finished editing the *passwd* file, unmount the Winchester disk root file system from the miniroot. Type:

   cd /

   This makes sure that your current working directory is not on the Winchester disk file system when you unmount the Winchester disk.

7. Type:

   /etc/umount /dev/dw00a

   This unmounts the Winchester disk from */mnt*.

8. Turn off the workstation.

9. Reset configuration switch 4 to up and switches 5 and 6 to autoboot (up, up). Remove any diskettes from the diskette drive.

10. Turn the workstation back on.

11. When the **login:** prompt appears, login as *root*. The system should not ask for a password.

12. When you get the # prompt, use the **passwd** command to assign a new root password.

As you can see, the diskette distribution allows access to root privileges without needing the root password. *Keep these diskettes in a secure place.*

Nonresponsive Terminal

If users encounter a nonresponsive terminal they will probably come to you for advice. A terminal may not respond for a number of reasons. Many of them are temporary problems that can be corrected by taking non-drastic action. The worst problem in this section can be dealt with by turning the workstation off then on again. Problems that need more drastic solutions are discussed in Section 8.

Problems that can cause a non-responsive terminal are:

- The pause character (most often ⟨CTRL-S⟩) was pressed.
- No paper in a terminal that requires paper.
- The port is not configured for login.
- Incorrect communications parameters.
- Hung process.

Try typing the "go-ahead" character (most often ⟨CTRL-Q⟩) to start communications again. This counteracts the action of the pause character.

If your terminal requires paper, make sure that there is paper available and properly installed. See the operator's manual for your terminal for instructions on installing paper.

Make sure that the port you are attaching the terminal to is configured for login. The Port Configuration menu of the sysadmin interface controls this parameter.

Incorrect Communications Parameters

Communications parameter on the terminal and on the workstation port that the terminal connects to must agree. If they don't, communications may be garbled, and may even be nonexistent.

Typically, the communications parameters for a terminal need only be set to agree with the workstation parameters once. Then, unless you change the configuration of the port the terminal is connected to, the communications between the terminal and the workstation should never alter.

Make sure the following communications parameters between the workstation port and the terminal are set to agree with each other:

- Terminal type — set the workstation port to agree with the type of terminal you have using the *sysadmin interface*. A list of acronyms for the more common terminals is available in Appendix D. If you do not find your terminal in this list, the file */etc/termcap* contains a list of all the terminals that the workstation recognizes.
- Baud rate — set the terminal to agree with the port setting from the *sysadmin interface*.
- Parity — should be even.
- Communications flagging — should be full duplex.

Check the configuration of the port you have the terminal connected to as opposed to the the configuration of the terminal. You can use the sysadmin interface to tell you what terminal type the system expects at the port, whether the port is configured for a login device (terminal), and what baud rate (configuration type) the port is set for. Section 4 discusses the sysadmin interface.

Check the operator's manual for your terminal for instructions on setting communications parameters for your terminal.

Hung Process

A *process* is a task performed by UTek. There can be more than one process spawned when a user enters a command, but there must be at least one process for each command that is currently running.

Each process has a process ID, which is a number that is assigned to the process when it is created. Process IDs go from 0 to 30000, then start over again.

It's possible for a process to stop before it should. This may happen if the user tries to interrupt the process at the wrong moment, or for a variety of other reasons, none common.

You should be able to terminate a process running in the foreground by pressing ⟨CTRL–C⟩ or ⟨CTRL–\⟩. If you press ⟨CTRL–\⟩, the process should stop and a *core dump* should occur. A core dump is a snapshot of the condition of the kernel for a given instant. Core dumps occur to facilitate debugging. The core dumps into the file *core* in the current directory. You can delete this file if you aren't in the process of debugging something.

A *hung process* is a process that won't terminate when you type either of these characters. It's possible to correct the hung process without turning the workstation off.

If the hung process is the only process currently running on the hung terminal, or if you or the user doesn't mind losing the other processes running on the terminal, turning the terminal off then on again or disconnecting the terminal from the communications port should kill all processes on the terminal, effectively logging out whoever is using the terminal. You can use this method on all terminals, even the console, that are connected to one of the workstation's RS–232–C port.

However, if you don't want log off the system or destroy other processes you may have running, you need to know the process ID for the hung process. You can get this number by running the **ps** utility.

If the problem is on a terminal that is not the console, run **ps** from the console. Log in as *root* and type:

ps –x –t *x*

Where *x* is the tty number of the terminal with the hung process. This number can be in one of two forms:

1. **tty***nn*

   where *nn* is the appropriate two–digit number.

2. *nn*

   where *nn* is the appropriate two–digit number.

For example, If you wanted to find out the processes for the terminal connected to standard RS–232–C port 0, either of the following command lines would tell you.

- **ps –x –t tty00**
- **ps –x –t 00**

This lists all the processes that were generated at the terminal with the problem. It's difficult to tell which process is causing the problem, so unless the user can help you pinpoint the offending process (the column labeled **COMMAND** in the list from the **ps** command is the utility that spawned the process), you should kill all the processes on that terminal. The user will have to log in again, but at least the terminal will be usable. Remember, as you gain more experience with the system, you will find it easier to identify hung processes, and you will be able to fine tune your approach to killing them off.

If the problem is on the console, move your base of operations to another terminal. Log in as *root*, then find the process number of the problem process by typing:

ps –x –t console

This lists all the processes that were generated at the console.

It's difficult to tell which process is causing the problem, so unless you can pinpoint the offending process (the final piece of information in the **ps** list is the utility that spawned the process), you should kill all the processes on that terminal.

If you are trying to help a user with a hung process, the user should try and help you identify the problem process. The user will have to log in again, but at least the terminal will be usable. Remember, as you gain more experience with the system, you will find it easier to identify hung processes, and you will be able to fine tune your approach to killing them off.

To kill processes, use the **kill** command.

You must be logged in as *root* to kill processes that you don't own.

Type:

kill *process_ID*

Where *process_ID* is the process ID of the process you are trying to kill. Do this for each process you want to kill.

Check the terminal. If it hasn't cleared, then the offending process wasn't killed. If this occurs, you must use a stronger version of the **kill** utility.

Type:

kill -15 *process_ID*

Kill -15 terminates the process with *process_ID* no matter what. Be very careful what processes you kill with **kill -15**. If you use this command on one of the important system processes, it halts the system without warning. See *kill(1)* for more information on the **kill** command.

Be sure to log out of the root account once you have killed the offending process or processes.

Hung Background Process

A process running in the background cannot hang the terminal. However, users may come to you and complain that the processes they put in the background aren't finishing.

If the process is running in the background, use the **kill** command. You can do this from the user's terminal if the process is running in the background or from the console if the process is running in the foreground, as described earlier. Users can use the **kill** command to kill any processes they own. It is a good idea for you to teach the users to be responsible for their own processes.

Use the **ps** command to determine the process number and the **kill** command to kill the process.

System Not Responding or Responding Slowly

There are various *system resources* that must be shared among all users on the system. If one or more processes are using more than their share of system resources, the performance of the entire system may be hampered.

The system resources that can affect system performance are:

CPU time

CPU time is the amount of time the workstation's Central Processing Unit spends on a process. The CPU must divide its time between all processes that are running on the system at a given time. The amount of CPU time a process takes is a good indicator of how much system resources the process is consuming. You can determine the amount of CPU time a process is taking by looking in the **TIME** column of the process list produced by the **ps** command.

Disk interrupts

A disk interrupt occurs when the system reads from or writes to the Winchester disk. The more disk reads and writes a program needs, the more time the system must spend processing these interrupts, and the longer other processes on the system must wait for attention.

I/O interrupts

An I/O (input/output) interrupt occurs when the system accesses a device connected to one of the workstation's RS-232-C or GPIB ports. These are *priority interrupts*. This means that the system must process them first, and therefore other system activities must wait.

Memory buffers

Memory buffers are system data structures in the workstation's RAM (Random Access Memory). They are allocated to the kernel and to users as needed. When users use memory buffers, data is swapped in and out of RAM from data devices like the Winchester disk and the diskette drive. Kernel requests for memory buffers have priority over user requests for memory buffers. Therefore, if the system allocates too many memory buffers to the kernel, the users cannot use the system effectively.

A good indicator of system performance is the system's *load average*. The load average tells you how many processes are waiting to be processed by the CPU. To find out what the load average on your system is, type **uptime**. You should find out the load average on the system if the system seems sluggish to you, or if users complain about system sluggishness. As you gain experience with the system, you will be able to tell when the system is not performing up to speed.

The **uptime** command reports the load average with three decimal numbers. These represent, from left to right, the load average within the last minute, the load average within the last 5 minutes, and the load average within the last 15 minutes. If the current load average is greater than 2 it is high. Check it again in a minute or two. If it remains high or rises higher, use the **ps** command to find out what processes are running on the system. If many processes that take a lot of resources are running, you can:

- Be patient and wait out the slow period.
- Use the **wall** command to request that all users do not put more load on the system until the load average drops. You can also use the **shutdown -k** command to make people think you are bringing the system down to single user (this command announces that you are bringing the system down, but doesn't really do it. See *shutdown(8)* in the *UTek Command Reference* manual for details).
- Use the **renice** (re-nice) command to reset the priority of some of the processes on the system to a lower priority. This takes some of the load off the system because it no longer has to parcel out system resources to as many processes at the same time.

  You must be logged in as *root* to renice any processes you don't own. Always ask the users who own the processes you want to "renice" before you do it. For more information on the **renice** command, see *renice(1)*.
- Kill some of the resource-intensive processes with the **kill** command. This is a very drastic step; use it only if you are sure that a process shouldn't be consuming all those system resources, or if the system is so bogged down that it can barely run.

You must be logged in as *root* to kill processes you don't own. Kill processes only after informing the users who own the processes. They may need the work desperately.

Beware of killing any process that has a **?** entry in the **TTY** column of the **ps** list These are system processes (the **?** indicates that they are not associated with a specific terminal, but are running independently of user interactions), and killing them can halt the system. The only such processes that it is safe to kill are duplicate versions of daemon processes. There is a discussion of daemon processes in Section 6 of this manual.

Should you accidently kill an important system process, and thereby cause a system halt, you can bring the system back up by turning the workstation off then on again.

The following causes can lead to insufficient system resources:

- Runaway processes
- Runaway network processes
- Many users logged in either on the system directly or over the network.
- Many resource–intensive applications running at the same time.

Runaway Processes

Processes can become *runaway processes.* A runaway process is a process that continues to use system resources (such as CPU time) long after it should have died. This can happen if a process gets into an infinite loop, if a process doesn't stop when it should, or for a variety of other reasons, none common, but none impossible.

As you gain experience with the system, you will get a feel for how long various processes should run before they are finished. Some processes should run the entire time the system is in operation, whether single user mode or multiuser mode. These processes are **init**, **/bin/sh**, **swapper**, and **pagedaemon**. Others should run at all times the system is in multiuser mode (normal operation). These include all daemon processes and **getty** processes for terminals where nobody is logged in.

Then there are the processes users start when they invoke UTek commands. Some take a long time to run, like the **find** command with a long search string. Others should be finished quickly, like processes generated by the **date** command.

You can tell a process is runaway if you know it shouldn't take much time, but successive **ps** commands show it consuming more and more CPU time.

Use the **kill** command to kill runaway processes.

Runaway Network Processes

If your workstation is on a busy network, it receives many communications packets. The more communications packets the workstation receives, the more memory buffers the kernel needs to process these packets.

Once a memory buffer is allocated to the kernel, it is not released back to the system. Therefore, if the system receives too many communications packets, so many memory buffers are captured by the kernel that users can no longer use the system. This usually only happens if a host on the network is sending out an abnormal amount of messages, or if the system has been up for a long time.

To find out how many memory buffers the network is using, type:

netstat -m

This tells you how many *mbufs*, or memory buffers, are allocated to the kernel for network use. Example 7-3 shows a sample result of this command.

Notice the second to last line of the report in Example 7-3. If this line shows 90 or more Kbytes allocated to the network, then network use of memory buffers is too high. To correct this, reboot the system (that is, turn the workstation off, then on again quickly).

If you find the problem of too many memory buffers being claimed for network use reoccurs often, and you cannot identify the problem host on the network, or attribute the problem to the system being up too long, contact your Tektronix Field Office.

```
165/224 mbufs in use:
  84 mbufs allocated to data
       27 mbufs allocated to socket structures
       37 mbufs allocated to protocol control blocks
       17 mbufs allocated to routing table entries
0/32 mapped pages in use
60 Kbytes allocated to network (34% in use)
0 requests for memory denied
```

Example 7-3. Sample netstat -m Report.

Out of Disk Space

If the system runs out of disk space, users see the a message to that effect on their terminals when they try to use more disk space:

If the Winchester disk gets too full, that is, within 10% of maximum capacity, users are not able to create new files or enlarge present files. Only root can create files, but anybody can delete files.

This extra 10% gives you enough disk space as root to perform the necessary tasks to reduce the amount of disk space in use. The **df** command tells you exactly how much disk space is in use.

Possible ways to reduce the amount of disk space in use include:

- Delete the files discussed earlier in this section under Disk Space.
- Ask users to delete unnecessary files.
- Back up the system then remove files that are not currently needed.

Do not remove any user's files without first asking the user's permission. Ask the users to remove their own files, instead.

If you run out of disk space often, you may need a bigger Winchester disk for your 6130 workstation. Contact your Tektronix Field Office to inquire about larger disks.

Out of Inodes

The number of inodes on the system determines the number of files allowed. If your system runs out of inodes, you cannot create any more files.

The number of inodes on a file system is determined when the file system is created. The number of inodes that comes on your 6130 workstation depends on the size of the Winchester disk. Table 7–3 shows the default number of inodes on the three sizes of Winchester disks possible on the 6130 workstation.

If the file system runs out of inodes, a message to that effect appears on the console.

If the file system runs out of inodes, you have to delete files to free inodes. Ask users to delete unnecessary files, and delete all the files you can that are listed earlier in this section under Disk Space.

If this problem reoccurs often, you must rebuild the file system to include more inodes. The **newfs** command rebuilds the file system (see *newfs(8)*). Rebuilding file systems is not in the scope of this manual. If you think you need to do this, contact your Tektronix Field Office.

Table 7–3
DEFAULT INODES

Disk Size	Number of Inodes
40 Mbyte	9984
80 Mbyte	28224
140 Mbyte	43264

8

System Halts

INTRODUCTION

Although system halts are rare, they can occur. This section deals with system halts: what causes them, how to track down the cause, how to correct the fault (if possible), and how to recover from the halt.

It's a good idea to read this section before a system halt occurs. That way you might notice strange occurrences before the system halts that may help determine the cause of the halt.

This section cannot cover everything that can cause the system to halt. If you suspect that your system was halted by something not covered in this section, contact your Tektronix Field Office.

HARDWARE PROBLEMS

This section covers causes of, and recovery from, the hardware errors that are the easiest to recognize and correct.

No Power

The workstation doesn't do anything when you press the start/stop switch.

Possible causes of no power are:

- The power cord is not properly connected.
- The start/stop switch is not in all the way.
- The line voltage is incorrect.

Recovery

Possible recovery methods for no power are:

- Check the power cord to make sure it's plugged in on both ends.
- Make sure you are pushing the start/stop switch in all the way (flush with the front panel); the green light on the switch should be on.
- Make sure that the line voltage is set to the correct voltage: 110 volts (domestic) or 220 volts (European) as appropriate for your electrical service.

Figure 8-1 shows the back of the workstation, including the location of the power cord plug and the line voltage indicator.

If you have performed all the listed solutions and the workstation still doesn't power up, contact your Tektronix Field Office.

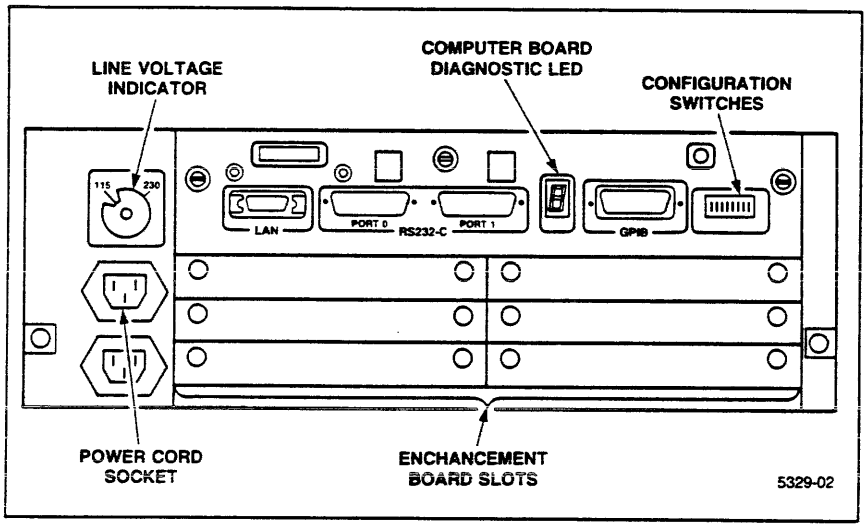


Figure 8-1. Back Panel of the 6130 Workstation

Option Board Failure

There are several option boards in the 6130 workstation. One of these option boards could fail. The 6130 has been designed so that it ignores failed option boards as much as possible. You can still run the workstation with a failed option board, but the system won't function as powerfully as it does with all boards working.

If one of the option boards does fail, the failure may take the one of the following forms:

- Messages that notify you of an option board or device failure appear on the console device.
- Devices connected to one of the RS-232-C ports, such as printers or terminals, may output unrecognizable or incorrect data. If the baud rate on the device is set to agree with the baud rate of the workstation, the option board for the port may be damaged.
- Devices may prohibit booting or workstation operation by constantly producing interrupts. Since you cannot get any response from the system if this is happening, this is the most difficult problem to diagnose.

You can also discover an option board failure when you start up the system. The workstation prints out on the console a list of the enhancement boards that it recognizes. If the start-up diagnostics were able to determine which board has failed, they pass this information on to the system, and the workstation does not recognize the devices on the failed board.

Recovery

If you discover a failure during operation, reboot the system by rapidly (within five seconds) turning the start/stop switch off then on again. If there are other users on the system, be sure to warn them that you are about to do this with the **wall** command. The start-up diagnostics determine which board has failed, if possible, and pass the information to the system, which then knows to ignore the devices on the failed board.

If you discover the failure at boot time, you do not need to reboot the system.

You might want to remove the failed option board. This doesn't fix the problem on the option board, but it keeps the problem from upsetting other parts of the system. Appendix B discusses removing option boards.

If you cannot boot the system, and you want to determine if one of the option boards is at fault, you must remove one option board at a time and try to boot the system with each option board absent. If the system boots when one board isn't installed, then you have correctly identified the problem. Appendix B discusses the procedure for removing option boards.

Contact your local Tektronix Field Office to get the problem option board fixed.

Main Computer Board Failure

Devices that reside on the main computer board could fail. The devices on the main board are:

- the diskette controller
- the Winchester disk controller
- the on-board RS-232-C devices
- the CPU (central processing unit)
- the MMU (memory management unit)
- on-board RAM
- the LAN (Local Area Network) controller
- the GPIB (General Purpose Interface Bus)

If any of these except the diskette controller, LAN, or GPIB fails, you can no longer use the system.

Failures of these devices and controllers are reported to the console device at boot time or during operation.

Recovery

Contact your local Tektronix Field Office.

Winchester Disk Problems

Your workstation comes standard with a Winchester disk. However, the 61TC01 streaming tape drive can also contain a Winchester disk. This discussion primarily covers Winchester Disk problems with the standard Winchester disk (in the 6130 cabinet), but many of the problems discussed can also apply to a 61TC01 optional hard disk. For information on formatting the 61TC01 optional hard disk, see Section 5.

The workstation's Winchester disk should start turning when you turn the workstation on. You should be able to hear a low hum that increases in pitch as the disk starts turning, then an intermittent soft rattling noise as the disk head reads or writes the disk. If the disk doesn't start turning, there is a hardware problem. Contact your local Tektronix Field Office.

Another hardware problem that can occur is bad blocks on the disk. You'll know that these are present because they produce

```
RD WDA HARD ERROR AT LOGICAL SECTOR xxxx
```

or

```
WR WDA HARD ERROR AT LOGICAL SECTOR xxxx
```

error messages, also known as *hard Read/Write* errors, on the console device. The *xxxx* represents a four-digit logical sector number. Write this number down. When you reformat your hard disk (covered later in this section) you can use this number to map out the defective sectors.

These errors can also be caused by the failure of the workstation's disk controller.

Recovery

If the workstation's Winchester disk doesn't start turning when you turn the workstation on, turn the power off and contact your local Tektronix Field Office.

If you get a hard read or write error, copy down the status information reported with the error and, if possible, note the file or command the system was accessing at the time of the error. Then, wait to see if another error occurs during normal operation. If none occurs, then the single error was probably just a temporary "glitch" on the disk.

If you are getting multiple hard Read/Write errors, start out by running the diagnostics on the Winchester disk, disk drive, and disk controller. See the information under Extended Diagnostics later in this section for information on how to boot the diagnostics operating system.

Run the tests on the Winchester disk drive and the disk controller. These tests tell you

- if your Winchester disk subsystem has errors
- what the errors are
- whether the error is on the board (controller), the disk, or the drive

See the *6130 System Diagnostics* manual for instructions on how to run the diagnostics and how to interpret the diagnostic messages.

If the diagnostics find errors in the disk drive or controller, or you are not able to run the disk diagnostics, contact your Tektronix Field Office.

If the diagnostics show errors on the disk itself, you should reformat the Winchester disk. Reformatting marks any bad blocks on the disk so that they are no longer used. When you reformat your hard disk (covered later in this section) you can use any bad sector numbers you received in an error message to map out the defective sectors.

Reformatting the Winchester Disk This discussion covers reformatting the Winchester disk located in the 6130 cabinet. A procedure in Section 5 provides you with information to reformat the 61TC01 optional hard disk.

CAUTION

Reformatting a hard disk should only be done by experienced system administrators. Reformatting a hard disk erases any data you have on that disk. Before you reformat the hard disk, you should back up any data contained on the hard disk to flexible diskettes, a 61TC01 streaming cartridge tape, or over the LAN.

Before reformatting the workstation's Winchester disk, back up as much of your data as possible to diskettes, to a cartridge tape, or over the LAN to another workstation. Do this because when you reformat the disk, all data on the disk is destroyed. If you have kept regular backups of your system, you can use these to restore your work after the disk and file system are rebuilt.

Use the following procedure to reformat the disk:

1. Turn the workstation off.
2. Insert the *standalone utilities* diskette (distributed with the workstation) into the diskette drive. Always keep this diskette in a safe place.
3. Set configuration switchs 5 down and switch 6 up to specify a diskette as the boot device. Figure 8-1 (earlier) shows the location of the configuration switches on the back of the workstation, and Figure 8-2 shows a detail of the switches in the position for normal operation.

   Table 8-1 shows how to set the configuration switches to select a boot device. Set configuration switch 4 to down. This lets you select the proper file on the diskette to boot from.

Table 8-1
BOOT DEVICE SETTINGS

Boot Device	Switch 5	Switch 6
Autoboot	up	up
Winchester disk	up	down
Diskette drive	down	up
LAN port	down	down

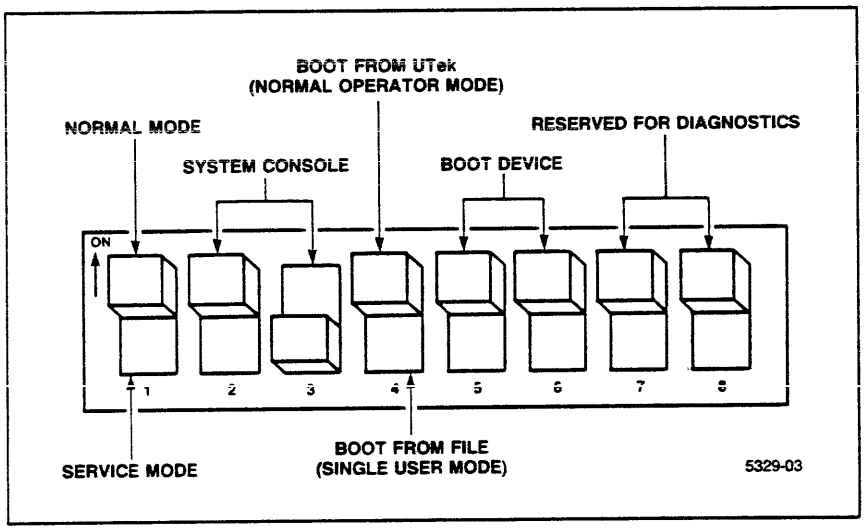


Figure 8-2. Configuration Selection Switches

4. Turn the workstation on. When the workstation prompts you with the character string >>>>, enter

   df(0,0)saformat

   and press <RETURN>. Do not enter a space between the right parentheses) and saformat. This loads the saformat program from the standalone utilities diskette, and provides you with the menu shown in Example 8-1. The *saformat* program lets you set parameters and it formats the disk. Formatting may take several minutes, depending on the size of the disk.

```
Drive Options

1) Quit
2) Winchester disk
3) Flexible diskette

Select by entering a number from 1 to 3:
```

Example 8-1. Saformat top-level menu.

5. Select item 2 from this menu. Once you have selected *Winchester disk*, the *saformat* program reads the identification header from your Winchester disk and displays it at the top of the screen (this is how the program knows what kind of disk you have), followed by the menu shown in Example 8-2.

```
Winchester Format Command Menu

 1) Return to the previous menu
 2) Quit the formatting program
 3) Select alternate disk drive

 4) Show maintenance directory
 5) Show maintenance allocation
 6) Modify formatting information
 7) Modify partition tables
 8) Modify maintenance tables

 9) Map out defective sectors
10) Sweep the disk for defects
11) Format the disk with the given information
12) Update maintenance data without formatting

Select by entering a number from 1 to 12:
```

Example 8-2. Winchester Format Command Menu.

6. If you are knowledgeable about the disk and the UTek file system, you can use the **saformat** to "tune" the disk. Items on the menu let you modify formatting information, partition tables, maintenance tables, or map out defective sectors (each of these items provide you with a menu that lets you change specific parameters). Two menu items show you the current maintenance directory and memory allocation.

   If your hard disk crashed, you should select item 10, Sweep the disk for defects, before doing the reformatting. This will check the disk for bad sectors. If you received an error message with information about bad sectors, and you have written this information down, use item 9 to map out these defective sectors.

   When ready to format the disk, choose the *Format the disk with the given information* option and pressing <RETURN>. If you have not changed any of the formatting values from their default (by selecting alternate menu choices), the default formatting values are used.

   The **saformat** program begins formatting the Winchester disk. The numbers that appear on the screen represent the disk cylinders as they are formatted. See Section 7 for a discussion of disk cylinders.

7. When the format is finished, the system informs you with the message:

   press RETURN to continue:

   When you press <RETURN>, the Winchester Format Command Menu is redisplayed.

8. Choose the option that quits the formatter and press <RETURN>. Answer y to the "do you really want to quit" question. You exit the formatter and the >>>> prompt reappears.

You must now rebuild the file system and restore the system to the disk. This procedure starts with installing the miniroot file system.

Installing the Miniroot File System The *miniroot* file system serves as a base for creating the real root file system. The following procedure details the steps involved in installing the miniroot system.

1. You should still have the standalone utilities diskette in the diskette drive when you begin the process of installing the miniroot system. The program on the standalone utilities diskette that copies the miniroot to the Winchester disk is called **sacopy**.

2. When you get the >>>> prompt after quitting the disk formatter, type:

   df(0,0)sacopy

   Do not enter a space between the right parentheses) and **sacopy**.

3. The program prompts you to make sure that the first volume (diskette) of the miniroot is in the diskette drive. remove the standalone utilities diskette, insert the miniroot diskette, volume 1, and press <RETURN>.

4. The program prompts you for the name of the device you want to copy *from*. Press <RETURN> here, as the device defaults to df(0,0), the diskette drive.

5. The program then prompts you for the device you want to copy *to*. Press <RETURN> here, as the device defaults to dw(1,0), the Winchester disk.

6. The program asks you to enter the block size and the number of records to copy. Press <RETURN> in answer to these questions; the block size defaults to 10240 and the number of records defaults to to 108.

   This begins loading the miniroot onto the Winchester disk's swap space. Dots are printed to the screen to indicate that the **sacopy** program is running.

7. When you see the message:

   ```
   Read n records from 1 volume(s)
   Insert next volume. Press RETURN to continue.
   ```

   remove volume 1 and insert the volume 2 miniroot diskette. Press <RETURN> to continue. The *n* in the message is the number of records copied so far from the miniroot diskette to the Winchester disk.

8. When you see the message in Step 7 again (except it informs you that it has read from 2 volumes), remove volume 2 and insert the volume 3 miniroot diskette. Press <RETURN> to continue.
9. When the copying of volume 3 is complete, the **sacopy** program asks you to insert the system diskette. Remove the volume 3 miniroot diskette, insert the miniroot system diskette, and press <RETURN>.

Once the miniroot system is loaded, you can install the regular *root* file system.

Installing the UTek Kernel At this point, the system is ready to load UTek. The following procedure details the steps involved in installing UTek once you have the miniroot installed.

1. When the **>>>>** prompt appears, type:

   df(0,0)vmunix

   Do not enter a space between the right parentheses) and **vmunix**. If you just press <RETURN> without entering anything here, the file that the system chooses automatically defaults to *vmunix*.

2. When the **sacopy** program asks you for a root device, type:

   dw00*

   This specifies the swap space, where the miniroot is.

You should then see the root prompt (#). This indicates that UTek is running in single user mode, and that you are logged in as *root* on the miniroot.

Now you must restore the system.

Rebuilding the Root File System Once the disk is formatted and the miniroot installed, you must rebuild the root file system before you can restore the system.

NOTE

You must have backup diskettes or cartridge tapes of your system that you made with the sysadmin interface or the **dump** *command to use this procedure. If you do not have backup tapes, see Appendix C.*

You are currently logged in to the miniroot with superuser privileges.

Using Backups from Streaming Cartridge Tape. The process for rebuilding the root file system assumes that the backups you want to restore are on diskette. If you want to restore from streaming cartridge tape, you must create the device file associated with the streaming cartridge tape drive. The drive must be installed on the workstation before you create the device files.

To create the device files for the cartridge tape, use the following procedure:

1. Move to the proper directory by typing:

   cd /dev

2. To create the device files, type:

   MAKEDEV tc*sd* **tc**

   where *s* is the slot number where the SCSI extension board is installed, and **d** is the drive number of the controller in the streaming cartridge tape drive.

   Remember that the drive number for 61TC01 tape drives is set to 4 at the factory. If you want to use another drive number, see the discussion of drive numbers in Section 5.

3. Return to the root directory of the miniroot by typing:

   cd

Running buildroot. To rebuild the root file system, use the **buildroot** program. This program rebuilds the root file system, restores your system from backup diskettes or tape, and restores the diagnostic operating system to the proper partition on the Winchester disk.

CAUTION

There is a space problem with miniroot and systems that have hard disks of 80 Mbytes or greater. See "Using Miniroot While Restoring Large Hard Disks" later in this section.

If you want to restore from backup diskettes, type:

/etc/buildroot

If you want to restore from backup cartridge tape, type:

/etc/buildroot -d /dev/tc

NOTE

You can exit the **buildroot** *procedure at anytime during its execution by pressing <CTRL-C>. Also, since* **buildroot** *is a shell script, you can customize it to specify different devices and file systems. Read the file* /etc/buildroot *to find out what exactly* **buildroot** *does, and edit the file (with* **ed**) *to change the actions of* **buildroot**. *Any modifications you make to* **buildroot** *are saved only until you reload the miniroot.*

The **buildroot** program is a shell script that calls a number of system utilities to rebuild the file system and restore your backups.

1. The first thing **buildroot** does is run **fsck** in preen mode to check the current condition of the root file system.

   Since you just reformatted the disk, the **fsck** fails and the following message appears:

   ```
   The root filesystem is corrupted.
   Construction of a new root filesystem is therefore
   necessary before any files can be restored.
   Do you wish to continue with this rebuild? [y,n](y)
   ```

   Type:

   y

   If you choose **n** here, the **buildroot** procedure terminates.

2. After you indicate that you want to continue with the procedure, **buildroot** runs **newfs**, which creates an empty root file system.

   If you see the following message, an error has occurred and **buildroot** terminates. Contact your Tektronix Field Office if you get this message.

   `newfs of` *filesystem* `failed. Quitting buildroot.`

   Filesystem is the disk partition where **newfs** is trying to rebuild the root file system.

3. After the root file system has been rebuilt, **buildroot** mounts the root file system onto the miniroot (*/mnt*) so that you can restore your backups.

   If you see the following message, either the file system is already mounted on the miniroot, or an error has occurred. In either case, **buildroot** terminates. Check if the file system is already mounted on */mnt*, and unmount it if it is, then restart **buildroot**. If the file system is not mounted on the miniroot and this message appears, contact your Tektronix Field Office.

   `mount of` *filesystem* `failed. Quitting buildroot.`

   Filesystem is the file system that **mount** is trying to mount onto */mnt*.

4. At this point, **buildroot** runs **restore**. The **restore** utility runs as many times as you need so that you can restore not only the latest level 0 backup, but all necessary incremental backups, too.

   Always restore the most recent level 0 backup first.

   Insert the first diskette or cartridge that holds the most recent level 0 backup into the appropriate drive and press <RETURN> when you see the message:

   ```
   Please be sure backup media is in the drive
   (press <RETURN> when ready):
   ```

   Media is **diskette**, unless you specified a streaming cartridge tape drive with the **-d** option.

   During the **restore**, an error message is displayed explaining that a file already exists for any files which already exist on the root filesystem which **restore** tries to install.

   Since **buildroot** already ran **newfs**, exactly one such message appears, indicating that */lost+found* already exists. This file is not overwritten, and the **restore** continues normally. Ignore the message.

   If you see the following message, an error has occurred and **buildroot** terminates. Contact your Tektronix Field Office if you get this message.

   ```
   restore from device failed. Quitting buildroot.
   ```

   Device is the device from which **restore** is trying to copy.

   When the level 0 restore is complete, the following message appears:

   ```
   Restore pass complete.
   ```

5. Once you have restored the level 0 backup, you must restore any incremental backups that were taken after the level 0 backup to get your system as close to the one you lost as possible.

   Buildroot runs **restore** over and over again until you tell it to stop, so that you can restore as many incremental backups as you need.

   Restore the most recent backup of each level unless you have a more recent backup of a lower level. Also, restore backups of different levels in order; level 1 backups first, then level 2 backups, and so on.

   For example, suppose you took five incremental backups since the level 0 backup you just restored. Further, suppose the first three of those incremental backups are level 9, the fourth one is level 5, and the last one is level 9 again. Use the next command line to restore the level 5 backup, then use it again to restore the last level 9 backup. It is unnecessary to restore the three earlier level 9 backups because the information in them is also in the level 5 backup.

   As another example, suppose you took four incremental backups since the level 0 backup you just restored, the first one being a level 5 backup, the next two being level 9 backups, and the fourth a level 4 backup. You need only restore the level 4 backup because it is more recent than all the others and of a lower level, and therefore includes the same data as the higher levels and earlier backups (as well as any data changed or added since those backups).

   The **buildroot** program prompts you with:

   ```
   Do you wish to restore from an(other) incremental backup? [y,n](n)
   ```

   If you have an incremental backup, remove the last level 0 backup diskette or tape, insert the first incremental diskette or tape into the drive, and type:

   y

   If you do not have an incremental backup to restore, continue with Step 7.

6. When the incremental backup has been restored, **buildroot** ask again:

```
Do you wish to restore from an(other) incremental backup? [y,n](n)
```

If you have another incremental backup to restore, remove the previous diskette or tape, insert the first diskette or tape of the new incremental backup into the drive, and type:

y

Do this step until all your incremental backups have been restored.

This procedure restores all files on all the backups, so if you had deleted some of the files after the last incremental backup, they are also restored, and you have to redelete them.

7. If you do not have another incremental backup to restore, remove the previous backup diskette or tape from the drive and type:

n

If you press <RETURN>, the answer defaults to n.

The **buildroot** program responds with:

```
Root file restoration complete.
```

8. After the root file system has been restored, **buildroot** restores the diagnostics operating system to the proper partition on the Winchester disk.

If you see the following message, **buildroot** is unable to install the diagnostics operating system:

```
bad /diags directory. Could not update diagnostics.
```

If you see this message, contact your local Tektronix Field Office.

9. After the diagnostics have been restored, **buildroot** unmounts the root file system from */mnt* on the miniroot with the **umount** command.

   If you see the following message, *umount* was unable to unmount the root file system.

   ```
   ** WARNING ** failed to unmount filesystem
   ```

   Filesystem is the file system that **umount** is trying to unmount from */mnt*. Ignore this message if it appears.

10. Finally, when the root file system has been unmounted, **buildroot** runs **fsck** in preen mode again to check the new root file system.

    If the **fsck** does not find errors, the **buildroot** finishes with the message:

    ```
    Root filesystem built.
    ```

    If the **fsck** finds errors it can correct, **buildroot** informs you that errors were found and corrected, then exits with the previous message.

    If the **fsck** finds errors it cannot repair, the following message appears:

    ```
    Filesystem is corrupted, please repair it.
    Exiting buildroot procedure.
    ```

    If you see this message, run **fsck** in interactive mode on the root file system as soon as you see the # prompt indicating that you are back in the miniroot. See Section 7 for details running on **fsck**, and Appendix A for a explanations of any **fsck** messages you receive.

    If you see the following message, contact your local Tektronix Field Office:

    ```
    Unknown error returned from fsck.
    ```

If You Do Not Have A System Backup. If you don't have a set of complete system backup diskettes (level 0 as produced by the **dump** command or the sysadmin interface backup function), you must load the operating system in from the copies of UTek available from Tektronix. Appendix C discusses the procedure for restoring the system if you didn't take backups.

Using Miniroot While Restoring Large Hard Disks

There is not enough space within miniroot to do a restore if the disk is 80 Mbytes or larger. The following procedure will show you how to workaround the problem.

1. Reformat your hard disk. Use the procedure found earlier in this section *Rebuilding The Root File System.*
2. Load the miniroot file system and run **newfs**. For more information about the command **newfs(8)**, see the *UTek Commands Volume 2* manual.
3. Mount the file system you just created on the */mnt* directory.
4. Make a temporary directory in */mnt*:

   ```
   mkdir /mnt/tmp
   ```

5. Remove miniroots */tmp* directory:

   ```
   rmdir /tmp
   ```

6. Link the temporary directory */mnt/tmp* to the miniroot file system:

   ```
   ln -s /mnt/tmp /tmp
   ```

7. Restore your disk.

By linking the temporary directory in */mnt* to the miniroot file system, enough space will be available for restore to write the files it requires when restoring.

After You Have Restored the System After the **buildroot** program is finished, the # prompt reappears.

Make sure that all the data you restored is properly written to the disk by typing:

```
sync
sync
```

Turn the workstation off and reset configuration switch 4 to up and configuration switches 5 and 6 to autoboot (up, up). Then turn the workstation on again to bring the system up normally using the new root file system. If you still encounter errors during normal operation, contact your Tektronix Field Office.

After the Reformat If you still encounter errors during normal operation, contact your Tektronix Field Office.

SOFTWARE PROBLEMS

This section covers causes of, and recovery from, the software errors that are the easiest to recognize and correct.

System Does Not Boot

The workstation may power up, but nothing happens after that. Two things are supposed to happen when the workstation is turned on: start-up diagnostics should run and the kernel should boot the system. If either of these two does not occur, UTek cannot begin normal operation.

Start-up Diagnostics Do Not Run

The area on the Winchester disk that stores the start-up diagnostics may become corrupted. If, when you turn the workstation on, the green light in the start/stop switch goes on and the disk starts turning, but nothing further happens, this may be the problem.

Recovery

Boot UTek from diskette.

1. Turn off the workstation.
2. Set configuration switches 5 and 6 to autoboot or boot from diskette (see Table 8-1).
3. Insert the miniroot system diskette into the diskette drive.
4. Turn on the workstation.

This should boot the system normally. Once UTek is running, you can restore the start-up diagnostics to the Winchester disk. To do this, run the **mkboot** program.

1. Log in as *root*.
2. Type: **/diags/utilities/mkboot -d /dev/dw00o -os /diags/diags_os 0**

This rewrites the diagnostics operating system to the disk partition where the start-up diagnostics expects to find it (dw00o).

Kernel Corruption

The system *kernel* is the central software for UTek that contains the information necessary for booting and running UTek. The difference between a corrupted kernel and other corrupted UTek software is that you can use the parts of UTek that are working even if some nonkernel software is corrupted, but none of UTek is usable if the kernel is corrupted.

Corruption of the system kernel can be caused by:

- A hardware problem in the disk subsystem.
- A software problem.
- The kernel file, */vmunix*, is missing or has been changed.

You can tell that the kernel is not operating correctly if the system seems to halt suddenly when you execute a command and if, at the same time the Computer Board Diagnostic LED on the workstation back panel (see Figure 8-1) isn't moving. This indicates that the workstation may be *compute-bound*, but not halted. A compute-bound machine has trouble responding to commands because its CPU (central processing unit) is too occupied to pay attention to incoming commands. If, in addition, the Computer Board Diagnostic LED on the back panel of the workstation is not moving and characters are not echoed when you type on the keyboard, the system has probably halted.

Recovery

Check to see if the kernel file, */vmunix*, is missing by listing the contents of the root directory, /, with the ls command. If *vmunix* is present, check if it has changed by looking at the last modification date. Use the ll command on the root directory, /, to see the modification dates of the files there. The last modification date of */vmunix* should be the date you first booted the system.

If *vmunix* is either missing or corrupted, you must install a new copy. If you can't get any response from the system, or if the ls utility doesn't work, install a new copy of *vmunix* anyway. It can't hurt, and it might help.

Installing a New Copy of /vmunix

Among the diskettes distributed with the system, the diskette labeled *UTek* contains a copy of the system kernel. If the system is already up, you can copy the kernel file *vmunix* from the miniroot system diskette. If the system is down, you can boot the system from the miniroot system diskette and then copy */vmunix* from your backup diskettes. These procedures are discussed here.

Installing /vmunix If the System is Up To install a new copy of *vmunix* if the system is already up:

1. Log in as *root*.
2. Insert the miniroot system diskette into the diskette drive.
3. Type:

   /etc/fsck -y /dev/rdf

   This checks the file system on the miniroot system diskette and corrects any file system corruption on the diskette.
4. Type:

   /etc/mount /dev/df *directory*

   Here, *directory* is a either an empty directory (no files), or a directory that you don't need to access while copying from the miniroot system diskette. (If the system is in single user mode, */usr* is a good directory to use.)

   This mounts the miniroot system diskette as a file system under the directory you specified.
5. Type:

   mv /vmunix /vmunix.old

   This moves the current kernel, */vmunix*, to another file, */vmunix.old*. Since the system is already running, you at know that the current kernel can at least boot the system. Saving it allows you to get the system started even if the new kernel you copy in is corrupted (you would do this by using the boot-from-file setting of switch 4).

6. Type:

   cp *directory*/**vmunix /vmunix**

   This copies *vmunix* from the miniroot system diskette to the Winchester disk.

7. Type:

   sync

   This ensures that the file is written to the Winchester disk.

8. Type:

   cd /

   This ensures that your current working directory is not on the diskette file system when you unmount the diskette.

9. Type:

   /etc/umount /dev/df

   This unmounts the miniroot system diskette from the directory you specified.

10. Remove the miniroot system diskette from the diskette drive.

Now a new copy of the kernel file *vmunix* is on the Winchester disk. At this point, reboot the system and run **fsck** to check for any other file system corruption. See Section 7 for more detail on **fsck**.

Installing /vmunix If the System is Down To install a new copy of */vmunix* if you cannot boot the system from the Winchester disk:

1. Turn the workstation off.
2. Make sure that configuration switches 5 and 6 specify autoboot or diskette (see Figures 8-1 and 8-2 and Table 8-1).
3. Insert the miniroot system diskette into the diskette drive.
4. Turn the workstation on. The workstation should boot up using the UTek diskette as its source.
5. Log in as *root*.

6. Type:

   /etc/fsck -y /dev/rdf

   This checks the file system on the miniroot system diskette and corrects any file system corruption on the diskette.

7. Type:

   /etc/mount /dev/df *directory*

   Here, *directory* is either an empty directory (no files), or a directory that you don't need to access while copying from the miniroot system diskette. (If the system is in single-user mode, */usr* is a good directory to use.)

   This mounts the miniroot system diskette as a file system under the directory you specify.

8. Type:

   cp *directory*/**vmunix /vmunix**

   This copies *vmunix* from the miniroot system diskette to the Winchester disk.

9. Type:

   sync

   This ensures that the file is written to the Winchester disk.

10. Type:

    cd /

    This ensures that your current working directory is not on the diskette file system when you unmount the diskette.

11. Type:

    /etc/umount /dev/df

    This unmounts the miniroot system diskette from the directory you specify.

12. Remove the miniroot system diskette from the diskette drive.

Now a new copy of the kernel file */vmunix* is on the Winchester disk. At this point, run **fsck** to check for any other file system corruption. See Section 7 under File System Maintenance for more detail on **fsck**.

If There Is More Than Kernel Damage If, after you replace the kernel file, you cannot run fsck, or if fsck cannot repair the damage to the file system, then you must restore the damaged files, possibly as many as all system files. It is unlikely that you'll have to restore more than a few system files.

The procedure for restoring files is discussed in Section 4 under File System Backup/Restore

Hardware–Caused Kernel Problems

If you have replaced the kernel and rebooted the system with the uncorrupted version of the kernel, and fsck either found no problems or fixed the problems it found, then any software kernel problems can be considered fixed. If you still have problems, there's probably something wrong with the workstation hardware causing kernel problems.

A way to pinpoint the problem is to remove the option boards one at a time. Then try to reboot the workstation with one board gone. If the system works with one of the option boards out, you have identified the problem. The procedure for removing option boards is discussed in Appendix B.

Contact your Tektronix Field Office when you have identified the defective board, or if you can't determine the cause of the kernel problems.

Missing or Corrupted System Files

The following files must be present on the disk and uncorrupted for you to bring the system up in single-user mode:

- */etc/init*
- */bin/sh*
- */dev/console*
- */dev/null*

In addition, the following files must be present on the disk and uncorrupted for you to bring the system up in multiuser mode:

- */etc/getty*
- */bin/login*
- */etc/ttys*
- */etc/passwd*

Recovery

If the system does not come up in single-user mode, but the hardware seems to be in working order, there is a good chance that one or both of the */etc/init* and */bin/sh* files is missing or corrupted. A message to this effect should appear on the console screen during the boot procedure. If this occurs, use the following procedure:

Booting the System from Diskette

1. Turn off the workstation.
2. Set configuration switches 5 and 6 to specify diskette (down,up). (See Figures 8-1 and 8-2 and Table 8-1.) Set configuration switch 4 to down. This lets you select a file on the diskette to boot from.
3. Insert the *standalone utilities* diskette into the diskette drive.
4. Turn on the workstation.
5. When the >>>> prompt appears, type

   df(0,0)sacopy

   Do not enter a space between the right parentheses) and **sacopy**.

6. The program prompts you to make sure that the first volume (diskette) of the miniroot is in the diskette drive. remove the standalone utilities diskette, insert the miniroot diskette, volume 1, and press <RETURN>.

7. The program prompts you for the name of the device you want to copy *from*. Press <RETURN> here, as the device defaults to df(0,0), the diskette drive.

8. The program then prompts you for the device you want to copy *to*. Press <RETURN> here, as the device defaults to dw(1,0), the Winchester disk.

9. The program asks you to enter the block size and the number of records to copy. Press <RETURN> in answer to these questions; the block size defaults to 10240 and the number of records defaults to to 108.

   This begins loading the miniroot onto the Winchester disk's swap space. Dots are printed to the screen to indicate that the **sacopy** program is running.

10. When you see the message:

```
Read n records from 1 volume(s)
Insert next volume. Press RETURN to continue.
```

    remove volume 1 and insert the volume 2 miniroot diskette. Press <RETURN> to continue. The *n* in the message is the number of records copied so far from the miniroot diskette to the Winchester disk.

11. When you see the message in Step 11 again (except it informs you that it has read from 2 volumes), remove volume 2 and insert the volume 3 miniroot diskette. Press <RETURN> to continue.

12. When the copying of volume 3 is complete, the **sacopy** program asks you to insert the system diskette. Remove the volume 3 miniroot diskette, insert the miniroot system diskette, and press <RETURN>.

When the miniroot system is installed, load the UTek system with the following procedure:

1. When the >>>> prompt appears, type:

   df(0,0)vmunix

   Do not enter a space between the right parentheses) and **vmunix**. If you just press <RETURN> without entering anything here, the file that the system chooses automatically defaults to *vmunix*.

2. When it asks you for a root device, type:

   dw00*

   This specifies the swap space, where the miniroot is.

You should then see the root prompt (#). This indicates that UTek is running in single user mode, and that you are logged in as *root* on the miniroot.

Before you use the **restore** command to copy the missing or corrupted file(s) into */etc/init* or */bin/shell* from your backup system to the Winchester disk, you must mount the file system on the Winchester disk.

1. Type:

   fsck -y /dev/rdw00a

   This checks the file system on the Winchester disk.

2. Type:

   mount /dev/dw00a /mnt

   This mounts the root file system from the Winchester disk onto the directory */mnt* on the miniroot.

3. If you have files that were dumped using the sysadmin backup interface, use the **restore** command to restore the missing or corrupted file(s) from your backup media. If you are not restoring from backup media, see Appendix C to find out how to Reinstall UTek.

   The device name for the diskette drive is **/dev/df**. The destination you should send the file */etc/init* to is **/mnt/etc/init** and y ou should send the file */bin/sh* to **/mnt/bin/sh**. (Remember, you mounted the regular root file system on the Winchester disk onto the directory */mnt*.) See **restore**(8) in the *UTek Command Reference* manual for details on the *restore* command.

4. When you are finished restoring the file(s) you need, unmount the Winchester disk root file system from the miniroot. Type:

   cd /

   This makes sure that your current working directory is not on the Winchester disk file system when you unmount the Winchester disk.

5. Type:

   /etc/umount /dev/dw00a

   This unmounts the Winchester disk from */mnt*.

6. Turn off the workstation.

7. Reset configuration switch 4 to up and switches 5 and 6 to autoboot (see Figures 8-1 and 8-2 and Table 8-1).

8. Turn the workstation back on. It should now be able to come up in multiuser (normal operating) mode.

If the system can come up in single user mode, but not multiuser mode, check to see if any of the four files */etc/getty*, */bin/login*, */etc/ttys*, or */etc/passwd* are missing or corrupted. There are a number of ways to correct the problem if any of the files are not usable.

The recommended method is:

1. Bring the workstation up in single-user mode:
   a. Turn off the workstation.
   b. Set configuration switch 4 to down.
   c. When you see the >>>> prompt, press <RETURN>.
   d. When the system asks you for a password, enter the *root* password.
2. When you get the # prompt, start the *nameserver* daemon by running the nameserver program. Do this by typing:

   /etc/nameserver

3. Use the su command to change yourself from root to your other login name.
4. Use the rcp command to copy the necessary file(s) from another node on the LAN to your workstation. You must have an account on the other node to use the rcp command. Do not run the rcp command from the remote node. See *rcp*(1) in the *UTek Command Reference* manual for details on the rcp command.
5. When you have copied the file(s) you need, turn off the workstation.
6. Set configuration switch 4 to up.
7. Turn the workstation back on. The system should come up in multiuser (normal operating) mode.

You can also use the method mentioned previously under the heading Installing the System from Diskette, that is, installing the miniroot system and loading the missing file(s) from your backup copy. Or, you can bring the system up in single user mode and copy the file(s) from the backup copy with the restore command.

Massive Root File System Corruption

Rarely, the root file system may become so corrupted that **fsck** can't repair it, or that in order to repair the file system, **fsck** must remove many important system files.

If you need to correct a lot of errors with **fsck**, and you find that normal operation has been curtailed by files being deleted by **fsck**, the file system was probably in this corrupted state.

Recovery

If this happens, you must rebuild the system, using the miniroot and your backups to get the closest to the system you lost.

The procedure for correcting massive file system corruption consists of loading the miniroot and using the **buildroot** procedure to rebuild the root file system.

Use the following procedure to load the miniroot:

1. Turn off the workstation.
2. Set configuration switches 5 and 6 to specify diskette (down, up) (see Table 8-1 and Figures 8-1 and 8-2). Set configuration switch 4 to down. This lets you select a file on the diskette from which to boot.
3. Insert the standalone utilities diskette into the diskette drive.
4. Turn on the workstation. When you get the >>>> prompt type:

   df(0,0)sacopy

   Do not enter a space between the right parentheses) and **sacopy**.
5. The program prompts you to make sure that the first volume (diskette) of the miniroot is in the diskette drive. remove the standalone utilities diskette, insert the miniroot diskette, volume 1, and press <RETURN>.
6. The program prompts you for the name of the device you want to copy *from*. Press <RETURN> here, as the device defaults to df(0,0), the diskette drive.
7. The program then prompts you for the device you want to copy *to*. Press <RETURN> here, as the device defaults to dw(1,0), the Winchester disk.
8. The program asks you to enter the block size and the number of records to copy. Press <RETURN> in answer to these questions; the block size defaults to 10240 and the number of records defaults to to 108.

   This begins loading the miniroot onto the Winchester disk's swap space. Dots are printed to the screen to indicate that the **sacopy** program is running.

9. When you see the message:

```
Read n records from 1 volume(s)
Insert next volume. Press RETURN to continue.
```

remove volume 1 and insert the volume 2 miniroot diskette. Press <RETURN> to continue. The *n* in the message is the number of records copied so far from the miniroot diskette to the Winchester disk.

10. When you see the message in Step 7 again (except it informs you that it has read from 2 volumes), remove volume 2 and insert the volume 3 miniroot diskette. Press <RETURN> to continue.

11. When the copying of volume 3 is complete, the **sacopy** program asks you to insert the system diskette. Remove the volume 3 miniroot diskette, insert the miniroot system diskette, and press <RETURN>.

Once the miniroot system is loaded, you can install the regular *root* file system.

Installing the UTek Kernel At this point, the system is ready to load UTek. The following procedure details the steps involved in installing UTek once you have the miniroot installed.

1. When the >>>> prompt appears, type:

df(0,0)vmunix

Do not enter a space between the right parentheses) and **vmunix**. If you just press <RETURN> without entering anything here, the file that the system chooses automatically defaults to *vmunix*.

2. When the **sacopy** program asks you for a root device, type:

dw00*

This specifies the swap space, on which the miniroot was loaded.

You should then see the root prompt (#). This indicates that UTek is running in single user mode, and that you are logged in as *root* on the miniroot.

Once the miniroot is loaded, you must restore the root file system and your data to the disk.

Rebuilding the Root File System Once the miniroot is installed, you must rebuild the root file system and restore the system.

NOTE

You must have backup diskettes or cartridge tapes of your system that you made with the sysadmin interface or the **dump** *command to use this procedure. If you do not have backup tapes, see Appendix C.*

You are currently logged in to the miniroot with superuser privileges.

Using Backups from Streaming Cartridge Tape. The process for rebuilding the root file system assumes that the backups you want to restore are on diskette. If you want to restore from streaming cartridge tape, you must create the device file associated with the streaming cartridge tape drive. The drive must be installed on the workstation before you create the device files.

To create the device files for the cartridge tape, use the following procedure:

1. Move to the proper directory by typing:

   cd /dev

2. To create the device files, type:

   MAKEDEV tc*sd* **tc**

   where *s* is the slot number where the SCSI extension board is installed, and **d** is the drive number of the controller in the streaming cartridge tape drive.

   Remember that the drive number for 61TC01 tape drives is set to 4 at the factory. If you want to use another drive number, see the discussion of drive numbers in Section 5.

3. Return to the root directory of the miniroot by typing:

   cd

Running buildroot. To rebuild the root file system, use the **buildroot** program. This program rebuilds the root file system, restores your system from backup diskettes or tape, and restores the diagnostic operating system to the proper partition on the Winchester disk.

If you want to restore from backup diskettes, type:

/etc/buildroot

If you want to restore from backup cartridge tape, type:

/etc/buildroot -d /dev/tc

NOTE

You can exit the **buildroot** *procedure at anytime during its execution by pressing <CTRL-C>. Also, since* **buildroot** *is a shell script, you can customize it to specify different devices and file systems. Read the file* /etc/buildroot *to find out what exactly* **buildroot** *does, and edit the file (with* **ed***) to change the actions of* **buildroot**. *Any modifications you make to* **buildroot** *are saved only until you reload the miniroot.*

The **buildroot** program is a shell script that calls a number of system utilities to rebuild the file system and restore your backups.

1. The first thing **buildroot** does is run **fsck** in preen mode to check the current condition of the root file system.

   There are two possible outcomes for this initial **fsck**.

   a. You may get either of the following two messages:

      ```
      The root filesystem structure is intact
      ```

      Or

      ```
      The root filesystem was corrupted, but has been repaired.
      ```

      Followed by:

      ```
      Creating a new root filesystem destroys any files
      left on the old system.  It is recommended, however, to
      ensure the integrity of the files about to be installed.
      Do you want to create a new root filesystem? [y,n](y)
      ```

This message indicates that the file system is in order according to **fsck**, but if you are going to restore the entire system, you should continue with **buildroot** and create the file system with **newfs** before you try to restore. If you don't do this, there may be problems with restoring all the files.

Type:

y

If you press <RETURN> here, the answer defaults to **y**.

b. Or, you could get either of the following messages:

```
The root file system is corrupted.
```

Or

```
Unknown fsck error
```

Followed by:

```
 The root filesystem does not appear to be usable.
Construction of a new root filesystem is therefore
necessary before any files can be restored.
Do you wish to continue with this rebuild? [y,n](y)
```

If you see this question, always elect to rebuild the root file system. Type:

y

If you choose **n** here, the **buildroot** procedure terminates.

2. After you indicate that you want to continue with the procedure, **buildroot** runs **newfs**, which creates an empty root file system.

   If you see the following message, an error has occurred and **buildroot** terminates. Contact your Tektronix Field Office if you get this message.

   `newfs of` *filesystem* `failed. Quitting buildroot.`

   Filesystem is the disk partition where **newfs** is trying to rebuild the root file system.

3. After the root file system has been rebuilt, **buildroot** mounts the root file system onto the miniroot (*/mnt*) so that you can restore your backups.

If you see the following message, either the file system is already mounted on the miniroot, or an error has occurred. In either case, **buildroot** terminates. Check if the file system is already mounted on */mnt* by trying to unmount it. If it unmounts without error, then it was mounted. Restart **buildroot**, the error should not reoccur. If the file system is not mounted on the miniroot and this message appears, contact your Tektronix Field Office.

```
mount of filesystem failed. Quitting buildroot.
```

Filesystem is the file system that **mount** is trying to mount onto */mnt*.

4. At this point, **buildroot** runs **restore**. The **restore** utility runs as many times as you need so that you can restore not only the latest level 0 backup, but all necessary incremental backups, too.

   Always restore the most recent level 0 backup first.

   Insert the first diskette or cartridge that holds the most recent level 0 backup into the appropriate drive and press <RETURN> when you see the message:

   ```
   Please be sure backup media is in the drive
   (press <RETURN> when ready):
   ```

   Media is **diskette**, unless you specified a streaming cartridge tape drive with the **-d** option.

   During the **restore**, an error message is displayed explaining that a file already exists for any files which already exist on the root filesystem which **restore** tries to install.

   If **buildroot** already ran **newfs**, exactly one such message appears, indicating that */lost+found* already exists. This file is not overwritten, and the **restore** continues normally. Ignore the message.

   If you did not let **buildroot** run **newfs**, there may be a number of the *file already exists* type of message. the **restore** does not overwrite these files.

   If you see the following message, an error has occurred and **buildroot** terminates. Contact your Tektronix Field Office if you get this message.

   ```
   restore from device failed. Quitting buildroot.
   ```

   Device is the device from which **restore** is trying to copy.

   When the level 0 restore is complete, the following message appears:

   ```
   Restore pass complete.
   ```

5. Once you have restored the level 0 backup, you must restore any incremental backups that were taken after the level 0 backup to get your system as close to the one you lost as possible.

   Buildroot runs **restore** over and over again until you tell it to stop, so that you can restore as many incremental backups as you need.

   Restore the most recent backup of each level unless you have a more recent backup of a lower level. Also, restore backups of different levels in order; level 1 backups first, then level 2 backups, and so on.

   For example, suppose you took five incremental backups since the level 0 backup you just restored. Further, suppose the first three of those incremental backups are level 9, the fourth one is level 5, and the last one is level 9 again. Use the next command line to restore the level 5 backup, then use it again to restore the last level 9 backup. It is unnecessary to restore the three earlier level 9 backups because the information in them is also in the level 5 backup.

   As another example, suppose you took four incremental backups since the level 0 backup you just restored, the first one being a level 5 backup, the next two being level 9 backups, and the fourth a level 4 backup. You need only restore the level 4 backup because it is more recent than all the others and of a lower level, and therefore includes the same data as the higher levels and earlier backups (as well as any data changed or added since those backups).

   The **buildroot** program prompts you with:

   ```
   Do you wish to restore from an(other) incremental backup? [y,n](n)
   ```

   If you have an incremental backup, remove the last level 0 backup diskette or tape, insert the first incremental diskette or tape into the drive, and type:

   y

   If you do not have an incremental backup to restore, continue with Step 7.

6. When the incremental backup has been restored, **buildroot** asks again:

```
Do you wish to restore from an(other) incremental backup? [y,n](n)
```

If you have another incremental backup to restore, remove the previous diskette or tape, insert the first diskette or tape of the new incremental backup into the drive, and type:

y

Do this step until all your incremental backups have been restored.

This procedure restores all files on all the backups, so if you had deleted some of the files after the last incremental backup, they are also restored, and you have to redelete them.

7. If you do not have another incremental backup to restore, remove the previous backup diskette or tape from the drive and type:

n

If you press <RETURN>, the answer defaults to **n**.

The **buildroot** program responds with:

```
Root file restoration complete.
```

8. After the root file system has been restored, **buildroot** restores the diagnostics operating system to the proper partition on the Winchester disk.

If you see the following message, **buildroot** is unable to install the diagnostics operating system:

```
bad /diags directory. Could not update diagnostics.
```

If you see this message, contact your local Tektronix Field Office.

9. After the diagnostics have been restored, **buildroot** unmounts the root file system from */mnt* on the miniroot with the **umount** command.

If you see the following message, *umount* was unable to unmount the root file system.

```
** WARNING ** failed to unmount filesystem
```

Filesystem is the file system that **umount** is trying to unmount from */mnt*. Ignore this message if it appears.

10. Finally, when the root file system has been unmounted, **buildroot** runs **fsck** in preen mode again to check the new root file system.

    If the **fsck** does not find errors, the **buildroot** finishes with the message:

    ```
    Root filesystem built.
    ```

    If the **fsck** finds errors it can correct, **buildroot** informs you that errors were found and corrected, then exits with the previous message.

    If the **fsck** finds errors it cannot repair, the following message appears:

    ```
    Filesystem is corrupted, please repair it.
    Exiting buildroot procedure.
    ```

    If you see this message, run **fsck** in interactive mode on the root file system as soon as you see the # prompt indicating that you are back in the miniroot. See Section 7 for details running on **fsck**, and Appendix A for a explanations of any **fsck** messages you receive.

    If you see the following message, contact your local Tektronix Field Office:

    ```
    Unknown error returned from fsck.
    ```

If You Do Not Have A System Backup. If you don't have a set of complete system backup diskettes (level 0 as produced by the **dump** command or the sysadmin interface backup function), you must load the operating system in from the copies of UTek available from Tektronix. Appendix C discusses the procedure for restoring the system if you didn't take backups.

After You Have Restored the System After the **buildroot** program is finished, the # prompt reappears.

Make sure that all the data you restored is properly written to the disk by typing:

```
sync
sync
```

Turn the workstation off and reset configuration switch 4 to up and configuration switches 5 and 6 to autoboot (up, up). Then turn the workstation on again to bring the system up normally using the new root file system. If you still encounter errors during normal operation, contact your Tektronix Field Office.

Disk Errors

If you get a single Read or Write error on the Winchester disk, copy down the information that comes with the error, and wait to see if it reoccurs.

These errors take the forms:

```
RD WDA HARD ERROR
```

or

```
WR WDA HARD ERROR
```

If you get multiple Read or Write errors when using the Winchester disk, follow the instructions for reformatting the disk under the heading Winchester Disk Problems earlier in this section.

RS-232-C Problems

If the response you get when you type on an RS-232-C terminal is not what you expect, there may be communications problems between the workstation and the terminal.

If the characters you type are not echoed back, you may have configuration problems.

If the characters you type are echoed twice, there may be parity problems.

If the characters that appear on the terminal screen are garbled, there may be baud rate or terminal type problems.

Recovery

Check the configuration of the port you have the terminal connected to as compared to the the configuration of the terminal. You can use the **sysadmin** interface to tell you what terminal type the system expects at the port, whether the port is configured for a login device (terminal), and what baud rate the port is set for. Section 4 discusses the **sysadmin** interface in detail.

Make sure the following communications parameters on the terminal are set correctly. The ones to watch for are:

- Baud rate — set this to agree with the port setting from the **sysadmin** interface.
- Parity — should be *even.*
- Communications flagging — should be *full duplex.*

Check the operator's manual for your terminal for instructions on setting these communications parameters for your terminal.

SYSTEM PANIC MESSAGES

Occasionally the system notifies you, by sending a *panic* message to the console device, of the cause of a sudden halt immediately before it happens.

The purpose of panic messages is to give you some idea of the reason for a catastrophic system failure. There is not much time for the system to display the message before the crash, so the information you get can be rather cryptic.

A properly operating system should never generate a panic message. If your system sends a panic message to the console device, copy it down, and contact your Tektronix Field Office immediately. You should also make carefully written notes on what the system was working on immediately before the panic message appeared and the system halted.

EXTENDED DIAGNOSTICS

The workstation has an extended diagnostics operating system that you can use to find problems in the workstation hardware.

To run the extended diagnostics operating system on the Winchester disk:

1. Set configuration switch 4 to down (see Figures 8-1 and 8-2).
2. When the system prompts you with >>>>, type:

   diags

At this point, the diagnostics operating system begins. See the *6130 System Diagnostics* manual for details on the diagnostics operating system.

9

System Reconfiguration

Concepts

Reconfiguring your workstation lets you modify the operating system to suit the needs of your particular site. Depending on the enhancement products you add to, or remove from, your system, the operating system kernel must change to recognize the enhancements. By supporting only the enhancement products installed on your system, you get greater system performance and flexibility.

To reconfigure your system, you run a program called **sysconf**. The **sysconf** program builds a new UTek kernel that supports only those enhancement products (devices) you request. By default, the **sysconf** programs builds a kernel that supports all the enhancement products installed on the current system. But you can also use **sysconf** to build kernels that are targeted for another system. The **sysconf** program also adjusts tuneable system parameters, including the time zone.

The **sysconf** program displays as a series of menus. The menus let you select the devices you want to include in the new UTek kernel. After you select the devices you want, **sysconf** creates new device driver tables and links them to a standard kernel.

The **sysconf** configuration software came on four of the nine diskettes you received with your workstation. These four diskettes come in the same diskette box as your miniroot and standalone utilities diskettes, and are labeled *System Configuration — 6100.*

The configuration software must be installed on your workstation before you can effectively use new enhancement products that you purchase or develop.

Overview

What is in the Configuration Software

The configuration software contains a standard kernel, the utilities, the libraries, and the *device description files* necessary to reconfigure your system. Each device description file gives the **sysconf** program information about a specific enhancement product.

You rarely need to use the configuration software (usually you only need to use it when you install an enhancement product that requires a device driver or if you want to tune your operating system parameters). Once you install the configuration software and use it to reconfigure your system, you can remove it from the hard disk until you need it again. This saves you 1.4 Megabytes of storage space, and you can always reinstall the configuration software from the diskettes when necessary.

When to Reconfigure Your System

You need to reconfigure your system when you:

- add an enhancement product
- remove an enhancement product
- create a custom kernel for use on a different system

Enhancement Products

Each time that you purchase enhancement products, you receive a tape or diskette that contains the device driver and additional information about the device. You must install the diskette or the tape before you can reconfigure your system to include the new enhancement. The tape or diskette that comes with the enhancement product is NOT the configuration software discussed in this section. It contains information that must be installed separately from the configuration software. See the installation manual for the enhancement product for instructions on installing the new device driver software.

The following enhancement products are available for your 6130 workstation. Support for these enhancements is available in the standard kernel.

- 61KR01 Dual RS-232-C Interface
- 61KR02 Synchronous/Asynchronous Interface
- 61KP01 Hard Copy Interface
- 61KP03 High Speed GPIB Interface
- 61KP04 SCSI Mass Storage Interface
- 61MP01/02/03 Expansion Memory
- 3F DMA Terminal Interface

Standard Devices for the 6100 Series

In addition to the enhancement products for the 6100 Series, some devices are standard on your workstation. These devices include:

- the diskette drive
- the hard disk drive
- the local area network port
- the GPIB interface

Because these devices are standard, you do not have to install their corresponding device drivers. But as you run the program that installs the device drivers for enhancement products, you see device listings for these standard devices. You have the option of removing support for the local area network and GPIB devices if they are not being used.

Installing Configuration Software

It is possible the configuration software may have already been installed and never removed. To find out if the configuration software is on your system, look for the */usr/sys/conf* directory. If you find this directory, look for the program **sysconf**. If **/usr/sys/conf/sysconf** exists on your workstation, you probably do not have to install it again, and you can proceed to the *Reconfiguring the System* portion of this section. If the */usr/sys/conf/sysconf* program or the */usr/sys/conf* directory does not exist, you must reinstall the configuration software. If you are in doubt, reinstall the configuration software to be safe. Installation takes about 15 minutes.

This overview summarizes the steps for installing configuration software on the 6130 workstation. Each step is presented in detail on the following pages.

1. Log in to the system as **root** and enter the **sysadmin** interface.

2. Insert the diskette into the diskette drive.

3. Install the software.
   - Select the Installation option on the System Administration Menu.
   - Select the Install Software option on the Installation Menu.
   - Specify the media type.
   - Respond to screen prompts as necessary.
   - Return to the Installation Menu.
   - Leave the sysadmin interface.

4. Remove the diskette from the diskette drive.

1. Log in to the System

To install software, you must be in the *sysadmin interface*. There are three ways to get into the sysadmin interface:

1. Log in as *sysadmin* and enter the sysadmin password when prompted.

2. Log in as *root* and type:

   /etc/sysadmin

3. Type:

   /etc/sysadmin

   from your regular account, and enter the sysadmin password when prompted.

2. Load the Diskette

After you have entered the sysadmin interface, load the first configuration software diskette (labeled **1** of **4**) into the diskette drive.

3. Install the Software

Select the Installation Option

The system displays the System Administration Menu (see Figure 9-1).

```
                                    Version 2.4

                System Administration

    1: (S)ystem Configuration Maintenance
    2: (F)ile System Backup/Restore
    3: (I)nstallation of Optional Software
    4: (U)ser Login Account Maintenance
    5: (G)roup Account Maintenance

    Enter ? for general help -->
    (Q)uit, <ESC> Back, (?,H)elp
```

Figure 9-1. System Administration Menu.

NOTE

If the version number of the System Administration Menu is higher than the one shown above, the menus you see may be slightly different.

Enter **3** to select Installation of Optional Software.

For online help, press **?**, followed by one of the option numbers (1-5). An explanation of the selected option displays.

Select the Install Software Option

The system displays the Installation Menu (Figure 9–2).

```
            Installation of optional software

      1: (I)nstall software
      2: (L)ist contents of cpio file

      Choose one of the above ->
      (Q)uit, <ESC> back, (?,H)elp
```

Figure 9–2. Installation Menu.

Enter **1**, to install software.

For on–line help, press **?** followed by one of the option numbers (1 or 2). An explanation of the selected option displays.

Specify the Media Type

At the bottom of the screen, the system prompts you to identify the software medium (Figure 9–3):

```
What medium is the software being installed from ?
        1) diskette
        2) 9-track magnetic tape
        3) streaming tape cartridge (QIC-24 format)
        4) file
        5) network

Enter number or <ESC> to go back->
```

Figure 9–3. Installation Menu — Bottom Portion.

Type the number that corresponds to the media type. Enter **1** to install from a diskette.

If you want to exit from the installation process at this point, press <ESC>. The system returns you to the top portion of the Installation Menu (Figure 9–1).

Respond to Screen Prompts as Necessary

The system displays these messages as it begins the installation process:

```
Searching for INSTALL procedure...
```

Then:

```
Found INSTALL procedure.
```

Then:

```
Please be sure the first volume is in the drive and then
type any key to continue or <q> to quit. -->
```

Press any key except **q** to continue.

The system may ask you again to verify that the source has been correctly loaded. It then displays the message:

```
Executing INSTALL procedure.
```

The system lists the names of the files as it moves them from the diskette to your Winchester disk, a process that takes several minutes. The system displays this message when it has finished transferring all the files from the first diskette:

```
No more data on this diskette.

To continue this installation, insert
the next diskette then press <RETURN>.

To quit, press <q> then press <RETURN>.

-->
```

To continue the software installation:

1. Remove the first diskette from the diskette drive.
2. Insert the next diskette into the diskette drive.
3. Press ⟨RETURN⟩.

The system continues to copy the software files from the diskette onto your Winchester disk.

To interrupt the installation process at this point:

1. Type **q** followed by ⟨RETURN⟩.
2. The system displays the message:

   ```
   Session terminated by user.
   ```

   and tells you the installation process is not complete. Press ⟨RETURN⟩. The system returns you to the shell prompt.

NOTE

This interruption means that the software installation has NOT been completed. To install the software at a later time, be sure your diskette is correctly loaded, then begin the installation procedure again from Step 1.

When all files have been copied from the source, the system lists the location of the newly-installed files. The programs and utilities are placed in the */usr/sys/conf* directory. The device description files are placed in the */usr/sys/conf/descrip* directory, and the name of each device description file has a *.d* suffix.

Because the installation procedure checks the RCS (Revision Control System) version numbers of the new files, the following message may display:

```
Ident of filename is same or older than existing version.
File will NOT be replaced.
```

If you are installing the configuration software on top of existing files, an existing files is not replaced if its version number is current. The message that displays is for your information; installation is proceeding normally.

The system then displays a message or series of messages to tell you that the installation process is complete.

If one of these messages displays:

```
***** SOFTWARE EXTRACTION FAILED *****
***** SOFTWARE INSTALLATION FAILED *****
      COMPLIANCE CHECK FAILED
```

the installation has not been completed properly. See section 5 of this manual or section 6 of the *6000 Family Software Installation* manual for more information.

Return to the Installation Menu

After the all the configuration software has been copied, press ⟨RETURN⟩ to return to the Installation Menu.

Leave the Sysadmin Interface

To leave the sysadmin interface, type **q**; the system asks the question:

```
Are you really sure you want to leave the sysadmin
interface (y/n)?
```

To answer YES, press **y** or **Y**.

The system returns you to where you were when you entered the sysadmin interface:

- If you logged directly into the sysadmin account, the system logs you out.

  Log into your own account, so you can verify your newly-installed software.

- If you logged in as *root*, the system displays the **#** prompt.

  To exit from *root*, press ⟨CTRL-D⟩. At the **login:** prompt, log into your own account.

- If you entered sysadmin from your own account, the system displays your usual system prompt.

4. Remove the Diskette

If you have not already done so, remove your diskette from the drive. Be sure to place the diskettes in their protective jackets and store them in a safe place.

After you have installed the configuration software, you are ready to use **sysconf** to:

- add an enhancement product
- remove an enhancement product
- create a custom kernel for use on a different system

USING THE CONFIGURATION SOFTWARE

Changing Kernel Options

By reconfiguring your system, you are building a new UTek kernel. When you build the kernel, you have the option of choosing the standard 6100 kernel. You get the standard 6100 kernel by default if you select the *Generate Kernel* menu item without specifically specifying which kernel you want.

Adding an Enhancement Product

Before you reconfigure your system for the enhancement product, you must complete the hardware installation of the enhancement. If the enhancement contains an external port, this may include loading the device driver from a diskette that came with the enhancement. See the installation manual that came with the enhancement product for details.

Once the hardware for the enhancement is installed and the login prompt for the workstation is displayed, you are ready to reconfigure the system and add support for the enhancement product. Complete the following steps:

1. Log in to the system as *root*.
2. At the root prompt (#), enter:

   cd /usr/sys/conf

3. Type **sysconf** at the prompt. The following menu (or a similar menu) displays:

```
        6000 Kernel Configuration Menu

    1) Display Configuration information
    2) Reinitialize Configuration information

    3) Modify device selection
    4) Tune Kernel parameters
    5) Generate Kernel

    6) Do makedev

    7) Quit the configuration program

Select by entering a number from 1 to 7:
```

Figure 9-4. Kernel Configuration Menu

4. Enter the number corresponding to the *Generate Kernel* menu item.

When you choose *Generate Kernel*, the **sysconf** program gathers information about what enhancement product hardware is installed on your system. Then it builds device driver tables for those enhancement products. **Sysconf** links the device driver tables to your UTek kernel. After linking the driver tables to the kernel, it creates a MAKEDEV utility that provides information for creating the special */dev* device files.

As **sysconf** generates the new kernel, the following messages display:

```
Created Assembler file param.s

Created System Definition File sysdef

Created MAKEDEV

as param.s -o param.o
ld -n -o vmunix -T 800 -x -e start 6X00.o param.o lib6X00.a
```

After these messages display, you return to the system prompt. There should be some new files in your current directory (*/usr/sys/conf*), one of which is *vmunix*. The *vmunix* file contains the system kernel. Go to the Booting the New Kernel part of this section, for instructions on installing the new kernel you just created.

Removing an Enhancement Product

If **sysconf** is not installed, you must install the configuration software (see the procedure earlier in this section).

There are two ways to remove support for an enhancement product from your system. You can remove the hardware for the enhancement and run **sysconf**. Or you can run **sysconf** and remove support for the enhancement by choosing the Modify Device Selection option at the Kernel Configuration Menu.

Removing the Hardware

To remove support for an enhancement by removing the hardware, complete the following steps:

1. Log in as **root**.
2. Type:

   shutdown *time*

   where *time* is **now** for immediate shutdown, *+minute* for shutdown in the given number of minutes, or *hh:mm* for shutdown at a specified time.
3. Press the front panel start/stop switch to the *stop* position. When the green light on the start/stop switch goes out, indicating power to the workstation is discontinued, continue with this procedure.
4. Remove the enhancement product hardware. See the enhancement product installation manual for instructions on how to do this.
5. Press the start/stop switch to the **start** position.

   A series of messages displays, describing the booting process. For more information about these messages, see the beginning of Section 5, System Start-up.
6. Run **sysconf** to generate a new kernel. Because the enhancement is removed, the kernel you create does not support the enhancement. Complete steps 1-4 in the Adding an Enhancement Product section to generate a new kernel.

Modify Device Selection

To remove support for an enhancement product by selecting devices in the **sysconf** menu, complete the following steps:

1. Log in to the system as **root**.
2. At the root prompt (#), enter:

   cd /usr/sys/conf

3. Type **sysconf** at the prompt. The following menu displays:

```
        6ØØØ Kernel Configuration Menu

    1) Display Configuration information
    2) Reinitialize Configuration information

    3) Modify device selection
    4) Tune Kernel parameters
    5) Generate Kernel

    6) Do makedev

    7) Quit the configuration program

Select by entering a number from 1 to 7: 3
```

Figure 9-5. Kernel Configuration Menu

4. Enter **3**, Modify device selection.

   The following menu displays:

```
                    Available Devices

1       KR02                            Synch/Asynch Ports
2       GPA [KP03]                      GPIB-A Port
3       DTI                             DMA terminal interface

4  Part 2:  Select drivers to be removed from kernel

Enter number of device to be supported in new kernel.
Select by entering a number from 1 to 4:
```

Figure 9-6. Available Devices Menu

NOTE

This menu may look different on your system, depending on what devices are available (see the beginning of this section). The number of available or selected devices is dynamic, so the number you enter to select a device changes as the menu changes.

Notice that the new GPIB-A driver is now the default driver. This new driver fixes many of the known problem that the old driver exhibited. However, if you want to continue to use the old driver, the old GPIB driver is now GPB. NOTE: only one GPIB driver is configurable at a time.

5. Enter the number that corresponds to *Select drivers to be removed from kernel.*

6. The following menu displays:

```
                Current Selected Devices

1       LNA                             Local Area Network
2       DF                              Floppy Disk Drive
3       DW                              Hard Disk Drive
4       PTY                             Pseudo Terminal Driver
5       RSA                             RS-232 Ports
6       KRØ1                            Option RS-232 Ports
7       KPØ4                            SCSI Interface
           Controlling:
8               TCA                     SCSI Cartridge Tape
9               DSA                     SCSI Disk
1Ø      GPB [KPØ3B]                     GPIB-B Port
11      KPØ1 [PPØ1]                     Hardcopy Interface

12    Return to main menu

Enter number of device for which to remove support in new kernel.
Select by entering a number from 1 to 12: 12
```

Figure 9-7. Current Selected Devices Menu

NOTE

This menu may look different on your system, depending on what devices are available (see the beginning of this section). The number of available or selected devices is dynamic, so the number you enter to select a device changes as the menu changes.

7. Enter the number that corresponds to the device you want to remove.

   To remove more than one device, when the menu displays again enter the number of the next device you want to remove. As you remove each device, the same menu displays again, except that the device you removed is no longer available and the numbering of each option has changed.

For example if you want to enable the old GPIB driver, enter number **2**. The following menu displays:

```
                    Available Devices
1       KR02                            Synch/Asynch Ports
2       GPB [KP03B]                     GPIB-B Port
3       DTI                             DMA terminal interface

4  Part 2:  Select drivers to be removed from kernel

Enter number of device to be supported in new kernel.
Select by entering a number from 1 to 4: 2

Disabling device GPA; only one GPIB device configurable at a time

Enabling device GPB
```

Figure 9-8. Available Devices Menu

When you have finished removing devices, enter the number that corresponds to *Return to main menu.*

8. The Kernel Configuration Menu (Figure 9-5) displays again. Enter **5**, Generate Kernel.

When you choose Generate Kernel, **sysconf** creates a new kernel. The following messages display:

```
Created Assembler file param.s

Created System Definition File sysdef

Created MAKEDEV

as param.s -o param.o
ld -n -o vmunix -T 800 -x -e start 6X00.o param.o lib6X00.a
```

After these messages display, you return to the system prompt. There should be a new file in your current directory (*/usr/sys/conf*), called *vmunix*. The *vmunix* file contains the system kernel. Go to the section entitled *Booting the New Kernel* for instructions on installing the new kernel you just created.

Building Kernels for Other Systems

If the configuration software is not installed, you must install it.

To select a kernel that supports particular devices, complete the following steps:

1. Log in to the system as *root*.
2. At the root prompt (#), enter:

   cd /usr/sys/conf
3. Type **sysconf** at the prompt. The following menu displays:

```
        6000 Kernel Configuration Menu

    1) Display Configuration information
    2) Reinitialize Configuration information
    3) Modify device selection
    4) Tune Kernel parameters
    5) Generate Kernel
    6) Do makedev
    7) Quit the configuration program
Select by entering a number from 1 to 7:
```

Figure 9-9. Kernel Configuration Menu

4. Enter **3**, Modify device selection.

5. The following menu displays:

```
                Available Devices

1       KR02                                Synch/Asynch Ports
2       GPA [KP03]                          GPIB-A Port
3       DTI                                 DMA terminal interface

4  Part 2:  Select drivers to be removed from kernel

Enter number of device to be supported in new kernel.
Select by entering a number from 1 to 4:
```

Figure 9-10. Available Devices Menu

NOTE

This menu may look different on your system, depending on what devices are available (see the discussion on enhancements earlier in this section). The number of available or selected devices is dynamic, so the number you enter to select a device changes as the menu changes.

6. Enter the number of the device you want to select. This device will be supported in the reconfigured system. To select none of the devices, enter the number corresponding to *Select drivers to be removed from kernel.*

   To select more than one device, when the menu displays again enter the number of the next device you want to select. As you select each device, the same menu displays again, except that the device you selected is no longer available and the numbering of each option has changed.

When you have finished selecting devices to be supported in the new kernel, enter the number corresponding to *Select drivers to be removed from kernel.*

The following menu displays:

```
                    Current Selected Devices

1       LNA                          Local Area Network
2       DF                           Floppy Disk Drive
3       DW                           Hard Disk Drive
4       PTY                          Pseudo Terminal Driver
5       RSA                          RS-232 Ports
6       KR01                         Option RS-232 Ports
7       KP04                         SCSI Interface
          Controlling:
8               TCA                  SCSI Cartridge Tape
9               DSA                  SCSI Disk
10      GPB [KP03B]                  GPIB-B Port
11      KP01 [PP01]                  Hardcopy Interface

12    Return to main menu
```

Figure 9-11. Current Selected Devices Menu

Enter number of device to remove support of device from new kernel. Select by entering a number from 1 to 12:

Figure 9-12. Current Selected Devices Menu

NOTE

This menu may look different on your system, depending on what devices are available (see the discussion on enhancements earlier in this section). The number of available or selected devices is dynamic, so the number you enter to select a device changes as the menu changes.

7. To include support for all the listed devices, enter the number corresponding to *Return to main menu.* The Current Selected Devices menu (Figure 9-10) shows you the devices supported in the standard kernel, devices that correspond to the hardware of your system, and those you selected from the Available Devices menu.

If you want to remove support for a device, enter the number that corresponds to that device. To remove support for more than one device, when the menu displays again, enter the number of the next device you want to remove.

When you have finished removing devices, enter the number corresponding to *Return to main menu.* All the devices that remain on the menu will be supported in the new kernel, and the system returns you to the Kernel Configuration Menu.

```
        6ØØØ Kernel Configuration Menu

    1) Display Configuration information
    2) Reinitialize Configuration information

    3) Modify device selection
    4) Tune Kernel parameters
    5) Generate kernel

    6) Do makedev

    7) Quit the configuration program

Select by entering a number from 1 to 7:
```

Figure 9-13. Kernel Configuration Menu

Enter **5**, Generate Kernel.

When you choose Generate Kernel **sysconf** creates a new kernel. The following messages display:

```
Created Assembler file param.s

Created System Definition File sysdef

Created MAKEDEV

as param.s -o param.o
ld -n -o vmunix -T 800 -x -e start 6X00.o param.o lib6X00.a
```

After these messages display, you return to the system prompt. There should be a new file in your current directory (*/usr/sys/conf*), called *vmunix*. The *vmunix* file contains the system kernel. Go to the next section, *Booting the New Kernel*, for instructions on installing the new kernel you just created.

BOOTING THE NEW KERNEL

After you run **sysconf** to create a new kernel, you must boot the new kernel. This procedure includes:

- making a back-up copy of the old kernel
- moving the new kernel to */vmunix*
- moving the new MAKEDEV program to */dev*
- shutting down the system
- booting the new kernel

1. Change directory to the root directory. Enter:

   cd /

2. Make a backup copy of the old kernel. Enter:

   mv /vmunix /vmunix.old

   If the new kernel does not work correctly, you can use this backup copy to boot the system. See your *System User's Guide*, section 4, Configuration Switches, for instructions on booting the system from the *vmunix.old* file.

3. Move the new kernel to the root directory. Enter:

   mv /usr/sys/conf/vmunix /vmunix

4. Make a backup copy of the MAKEDEV script.

   cp /dev/MAKEDEV MAKEDEV.old

5. Move the new MAKEDEV script to the */dev* directory. Enter:

   mv /usr/sys/conf/MAKEDEV /dev/MAKEDEV

6. To shut the system down enter:

   shutdown -r *time*

   where *time* is **now** for immediate shutdown, *+minute* for shutdown in the given number of minutes or *hh:mm* for shutdown at a specified time. The system reboots automatically.

   Instead of using **shutdown**, you can cycle the power by pressing the start/stop switch off, then quickly on again. The system reboots automatically.

   Messages display telling you that the system is going down.

In this manual, see the beginning of Section 5, System Start-up, for a description of what happens when the system reboots. As the new kernel is installed, particular messages display. One of the messages is:

```
booting /vmunix
```

Following the boot message, a list of devices supported by the new kernel displays. The devices that you specified in **sysconf** are included in that list, because they are now supported in the new kernel.

Running MAKEDEV

After you reboot the new kernel, you must run MAKEDEV to create the special files necessary to access the device. See Section 5 of this manual or the enhancement product installation manual for instructions on running MAKEDEV.

TUNING SYSTEM PARAMETERS

Why Tune Parameters?

In addition to configuring your workstation for different enhancement products, the System Configuration Package lets you adjust tuneable system parameters. Adjusting these parameters can modify the performance of your system to your particular needs. For example, if you are not using networking, you do not need as much process space in the kernel.

Maximum System Load

Unless you are very knowledgeable about the UTek kernel, the only system parameter you should tune is the system load. The exception to this is setting the time zone. It is not difficult to use **sysconf** to set the time zone, and you should do so when necessary.

The *maximum system load* is set to an integer value that approximates the number of users on the system. The default value of the maximum system load is 4, assuming a maximum number of four users on a workstation. The default value assumes that those four users are running processes that consume an average amount of computing resources. But if all the users are running processes that consume a large amount of CPU time, you may want to increase the value of the maximum system load.

When you increase or decrease the value of the maximum system load, the size allotted for the tables in the kernel increases or decreases. As the load changes, it automatically changes *all* the tuneable system parameters. You only need to change the value of the maximum system load, unless you want to change the values of particular parameters relative to each other.

Begin at the top **sysconf** menu (refer back to Figure 9-5), as explained earlier in this section.

To change the maximum system load, enter **4**, Tune Kernel parameters, at the main **sysconf** menu. The following prompt displays:

```
Is networking being used? [y,n](n)?
```

If your workstation is on a LAN, enter **y**.

If your workstation is not on a local area network, press <RETURN> or enter **n**.

Depending on whether or not your workstation is on a local area network, **sysconf** changes the value of some tuneable system parameters.

After you respond to the networking prompt, a prompt that lets you change the maximum system load displays:

```
What is the maximum load the kernel will handle?
maxload is currently set to 4
new value = (4):
```

To retain the maximum system load at 4 users, press <RETURN>.

To change the maximum system load, enter an integer from 1 to 6, depending on the maximum number of "average" users the system must support. If you try to enter an integer greater than 6, the following message displays:

```
WARNING:  it is not recommended to set maxload above 6
Are you sure that you want to set maxload to n? [y,n](n)
```

Press <RETURN> or enter **n** to choose another value for the maximum system load. You are prompted to enter a new value.

Enter **y** if you are sure that you want to set the maximum system load greater than 6.

NOTE

A maximum system load greater than six users can seriously degrade the performance of the workstation.

To cancel any changes you make, enter **8** at the Tuneable Kernel Parameter menu to return the parameters to their default values.

Other Tuneable Parameters

All the tuneable parameters are affected by changing the maximum system load and specifying whether or not your workstation is on a network. After you set the maximum system load, press <RETURN> to continue. The Tuneable Kernel Parameter menu displays. To see all the tuneable parameters, enter **9**, Display Tuneable Parameters, at the main menu.

Unless you want to change the values that a specific parameter has relative to other parameters, you do not need to reset other parameters. The exception to this is the time zone information.

Following are descriptions of tuneable parameters you can set from the Tuneable Kernel Parameter menu. To cancel any changes you make, enter **8** at the Tuneable Kernel Parameter menu to return the parameters to their default values.

Time Zone Information

The time zone information specifies the number of minutes west of Greenwich time (for example, 480 for Pacific Standard Time). If you have a European time zone, you can also set negative values for minutes east of Greenwich time. You can also set the type of daylight savings time being used: none, USA, Australia, Western Europe, Middle Europe, or Eastern Europe.

Process Limits

Use this menu to set the number of processes, the number of page maps, the number of text segments, and the total number of segments.

File I/O Limits

The file input/output limits include the number of **cd** commands across the Distributed File System, the maximum number of open files, and the number of inode structures.

General I/O Limits

The general input/output limits include the length of the time-out queue, the number of terminal character lists available, and the number of message buffers available.

Set Up Mass Storage Devices

Use this menu to set the default *root* device, the default argument list device, the default dump device, and the kernel dump limit.

Override Dynamically Set Kernel Parameters

Use this menu to override values that the kernel sets while it is running. These values include the number of pages of buffer space, the number of buffer headers, and the number of swap buffers. Dynamically set parameters are set by the kernel based on the amount of memory in the system at boot time. We strongly advise that you NOT adjust these parameters.

HOW SYSTEM CONFIGURATION WORKS

This part of this section is for users who are writing device drivers for installation using **sysconf**. It also provides more detail on where the package gets its system information.

Overview

The System Configuration package provides the user with an easy way to enter information that is necessary to configure the system. Internally, the System Configuration package provides access to system tables, specific information about devices, and a way of linking device driver tables to the kernel.

The actions performed by the System Configuration package can be broken down into the following steps:

- defining the system
- creating the device driver table
- assembling and loading the driver table

This section discusses those steps. It also discusses how you can make a new device available to the System Configuration package.

Defining the System

Before you build a kernel, you need to define the components of the target system. When you invoke **sysconf**, it queries the operating system tables to determine what peripherals and enhancements are attached to the workstation. When you generate a kernel, a system definition file called *sysdef* is created. The *sysdef* file defines the currently selected devices and tuneable parameters. At the **sysconf** main menu you can select **1** to see what devices are currently configured in the *sysdef* file.

If you invoke **sysconf** with the **-s** option, you can specify an alternate system definition file. For example, you could save system definition files from a previous **sysconf** and use them to build custom systems. For information on writing your own system definition file, see your *UTek Command Reference, sysdef(5)*. When you are using the **sysconf** menus, you can return the system definition file to its default value by entering option **2** at the Kernel Configuration menu.

Creating Device Driver Tables

The actual device driver program resides in a library of archived devices, */usr/sys/conf/dev.a*. In this discussion, the *device driver table* consists of the assembler file *param.s* created when you run **sysconf**.

The unique portion of the *param.s* file comes from the *device description file*. The device description file resides in */usr/sys/conf/descrip/*.d*, where * is the signature name of the device. The file contains information necessary to configure a device, including device driver tables, the name of the kernel **attach** routine, and a shell program that **MAKEDEV** can use to make special devices.

The device driver tables that become entries in the *param.s* file include device character and block switch. The *param.s* file also contains a table with peripheral device signature register values and their associated **attach** routine addresses.

When the user selects **6** at the main **sysconf** menu, the *param.s* file and a *MAKEDEV* file are created. The *MAKEDEV* file contains a shell program with specific information for creating the selected devices.

Assembling and Loading the Device Driver Table

When the device driver table, *param.s*, has been created, **sysconf** invokes a shell program to assemble *param.s* and load the following object modules:

- device driver table, *param.o*
- kernel, *6100.o* (or *6100_BSD.o* for 4.2BSD kernel)
- device driver library, *lib6X00.a*

These object modules are combined to form a new kernel, *vmunix*, that supports the device driver table created by **sysconf**.

Adding a New Device Driver

When you write a new device driver for your 6130 workstation, the System Configuration package provides an easy way to install the driver.

To give the System Configuration package the information particular to your device driver, you must write a device description file. See your *UTek Command Reference, devdes(5)*, for instructions on writing a device description file. You can also use the device description files in */usr/sys/conf/descrip* on your system as an example.

Once you have written the device description file, add your device driver to the archived library of device drivers using the **ar** utility:

ar r lib6100.a *yourdriver.o*

This command adds your device driver to the library of 6100 Series device drivers. The **ar** command calls **ranlib**, which converts the archive into a form that is more easily loaded by adding a symbol definition table at the beginning of the archive.

The device libraries are in */usr/sys/conf*. The kernel objects are named *6100.o* and *6100_BSD.o*. The device libraries are named *lib6100.a* and *lib6100_BSD.a*. *6100.o* and *lib6100.a* make the enhanced VM kernel, which is the standard 6100 kernel.

A

Appendix — Fsck Messages

This appendix is an explanation of the messages that you can get when using **fsck**. Except for some of the messages from the Initialization phase, these aren't errors in the use of the command, but messages that indicate the problems that **fsck** finds as it goes through the file system.

If **fsck** finds problems in the file system, it reports the error condition to you. If a response is required, **fsck** prints a prompt message and waits for a response. This section explain the meaning of each message, the possible responses, and the related messages. The first time you run through **fsck**, answer **no** or **n** to any such questions (except those that ask you if you want to continue checking the file system; answer "yes" to those). Then, the second time through, you can answer the questions **yes** or **no** depending on what you discovered about the file system damage on the first run through.

The messages are organized by the phase of the **fsck** program in which they can occur. The messages that can occur in more than one phase are discussed in Initialization.

Initialization

***C* option?**
Terminal. *C* is not a legal option to **fsck**. Legal options are —b, —y, —n, and —p. See *fsck*(8) in the *UTek Command Reference* for further details.

cannot alloc NNN bytes for blockmap

cannot alloc NNN bytes for freemap

cannot alloc NNN bytes for statemap

cannot alloc NNN bytes for lncntp
Terminal. **Fsck**'s request for memory for its virtual memory tables failed. Check disk usage and make some space on disk. Try **fsck** again. Possible hardware error.

Can't open checklist file:* *F
Terminal. The file system checklist file *F* (usually */etc/fstab*) can not be opened for reading. Check protection modes of *F*.

Can't stat root
Terminal. **Fsck**'s request for statistics about the root directory "/" failed. This should never happen. Contact your Tektronix Field Office.

Can't stat* *F

Can't make sense out of name* *F
Fsck's request for statistics about the file system *F* failed. It ignores this file system and continues checking the next file system given. Check protection modes of *F*.

Can't open* *F
Fsck's attempt to open the file system *F* failed. It ignores this file system and continues checking the next file system given. Check protection modes of *F*.

***F*: (NO WRITE)**
Either the —n flag was specified or **fsck** 's attempt to open the file system *F* for writing failed. All messages are printed out, but no modifications are attempted to fix file system corruption.

file is not a block or character device; OK
You have given **fsck** a regular file name by mistake. Check the type of the file specified.

Possible responses to the OK prompt are:

YES Ignore this error condition.

NO Ignore this file system and continue checking the next file system given.

One of the following messages appears:

MAGIC NUMBER WRONG

NCG OUT OF RANGE

CPG OUT OF RANGE

NCYL DOES NOT JIVE WITH NCG*CPG

SIZE PREPOSTEROUSLY LARGE

TRASHED VALUES IN SUPER BLOCK
Followed by the one of these messages:

F: ***BAD SUPER BLOCK:*** *B*

USE -b OPTION TO FSCK TO SPECIFY LOCATION OF AN ALTERNATE

SUPER-BLOCK TO SUPPLY NEEDED INFORMATION; SEE fsck(8).
The original copy of the superblock is corrupted. Select an alternative copy of the superblock from among those listed by **newfs** when the file system was created. For file systems with a blocksize less than 32K, specifying **-b 32** is a good first choice.

INTERNAL INCONSISTENCY: *M*
Fsck has had a fatal internal error with the message *M*. This should never happen. Contact your Tektronix Field Office.

CAN NOT SEEK: BLK B (CONTINUE)

Fsck's attempt to move to a specified block number *B* in the file system failed. This should never happen. Contact your Tektronix Field Office.

Possible responses to the CONTINUE prompt are:

YES Attempt to continue to run the file system check. Often the problem will persist. This error condition will not allow a complete check of the file system. A second run of **fsck** should be made to re-check this file system. If the block was part of the virtual memory buffer cache, **fsck** will terminate with the message "Fatal I/O error".

NO Terminate **fsck**.

CAN NOT READ: BLK B (CONTINUE)

Fsck's attempt to read a specified block number *B* in the file system failed. This should never happen. Contact your Tektronix Field Office.

Possible responses to the CONTINUE prompt are:

YES Attempt to continue the file system check. Often the problem will persist. This error condition will not allow a complete check of the file system. A second run of **fsck** should be made to re-check this file system. If the block was part of the virtual memory buffer cache, **fsck** will terminate with the message "Fatal I/O error".

NO Terminate **fsck**.

CAN NOT WRITE: BLK B (CONTINUE)

Fsck's attempt to write to a specified block number *B* in the file system failed. The disk is write-protected. Contact your Tektronix Field Office

Possible responses to the CONTINUE prompt are:

YES Attempt to continue to run the file system check. Often the problem will persist. This error condition will not allow a complete check of the file system. A second run of **fsck** should be made to re-check this file system. If the block was part of the virtual memory buffer cache, **fsck** will terminate with the message "Fatal I/O error".

NO Terminate **fsck**.

Phase 1 — Check Blocks and Sizes

CG C: BAD MAGIC NUMBER

The magic number of cylinder group *C* is wrong. This usually indicates that the cylinder group maps have been destroyed. This error marks the cylinder group as needing reconstruction.

UNKNOWN FILE TYPE I = I (CLEAR)

The mode word of the inode *I* indicates that the inode is not a special block inode, special character inode, socket inode, regular inode, symbolic link, or directory inode.

Possible responses to the CLEAR prompt are:

YES — De–allocate inode *I* by zeroing its contents. This will always invoke the **UNALLOCATED** error condition in Phase 2 for each directory entry pointing to this inode.

NO — Ignore this error condition.

LINK COUNT TABLE OVERFLOW (CONTINUE)

An internal table for **fsck** containing allocated inodes with a link count of zero has no more room. Recompile **fsck** with a larger value of MAXLNCNT.

Possible responses to the CONTINUE prompt are:

YES — Continue with the program. This error condition will not allow a complete check of the file system. A second run of **fsck** should be made to re–check this file system. If another allocated inode with a zero link count is found, this error condition is repeated.

NO — Terminate **fsck**.

B BAD I = *I*

Inode *I* contains block number *B* with a number lower than the number of the first data block in the file system or greater than the number of the last block in the file system. This error condition may invoke the **EXCESSIVE BAD BLKS** error condition in Phase 1 if inode *I* has too many block numbers outside the file system range. This error condition will always invoke the **BAD/DUP** error condition in Phases 2 and 4.

EXCESSIVE BAD BLKS I = I (CONTINUE)

There is more than a tolerable number (usually 10) of blocks with a number lower than the number of the first data block in the file system or greater than the number of last block in the file system associated with inode *I*.

Possible responses to the CONTINUE prompt are:

YES Ignore the rest of the blocks in this inode and continue checking with the next inode in the file system. This error condition will not allow a complete check of the file system. A second run of **fsck** should be made to re-check this file system.

NO Terminate **fsck**.

B ***DUP I = I***

Inode *I* contains block number *B* which is already claimed by another inode. This error condition may invoke the **EXCESSIVE DUP BLKS** error condition in Phase 1 if inode *I* has too many block numbers claimed by other inodes. This error condition will always invoke Phase 1b and the **BAD/DUP** error condition in Phases 2 and 4.

EXCESSIVE DUP BLKS I = I (CONTINUE)

There is more than a tolerable number (usually 10) of blocks claimed by other inodes.

Possible responses to the CONTINUE prompt are:

YES Ignore the rest of the blocks in this inode and continue checking with the next inode in the file system. This error condition will not allow a complete check of the file system. A second run of **fsck** should be made to re-check this file system.

NO Terminate **fsck**.

DUP TABLE OVERFLOW (CONTINUE)

An internal table in **fsck** containing duplicate block numbers has no more room. Recompile **fsck** with a larger value of DUPTBLSIZE.

Possible responses to the CONTINUE prompt are:

YES Continue with the program. This error condition will not allow a complete check of the file system. A second run of **fsck** should be made to re-check this file system. If another duplicate block is found, this error condition will repeat.

NO Terminate **fsck**.

PARTIALLY ALLOCATED INODE I = I (CLEAR)
Inode *I* is neither allocated nor unallocated.

Possible responses to the CLEAR prompt are:

YES De–allocate inode *I* by zeroing its contents.

NO Ignore this error condition.

INCORRECT BLOCK COUNT I = I (X should be Y) (CORRECT)
The block count for inode *I* is *X* blocks, but should be *Y* blocks.

Possible responses to the CORRECT prompt are:

YES Replace the block count of inode *I* with *Y*.

NO Ignore this error condition.

Phase 1B: Rescan for More Dups

B DUP I = I
Inode *I* contains block number *B* that is already claimed by another inode. This error condition will always invoke the **BAD/DUP** error condition in Phase 2. You can determine which inodes have overlapping blocks by examining this error condition and the DUP error condition in Phase 1.

Phase 2 — Check Pathnames

ROOT INODE UNALLOCATED. TERMINATING.

Terminal. The root inode (usually inode number 2) has no allocate mode bits. This should never happen.

NAME TOO LONG F

An excessively long path name has been found. This is usually indicative of loops in the file system name space. This can occur if the superuser has made circular links to directories. Remove the offending links.

ROOT INODE NOT DIRECTORY (FIX)

The root inode (usually inode number 2) is not directory inode type.

Possible responses to the FIX prompt are:

YES — Reassign the root inode to be a directory type. If the root inode's data blocks are not directory blocks, a VERY large number of error conditions will be produced.

NO — Terminate **fsck**.

DUPS/BAD IN ROOT INODE (CONTINUE)

Phase 1 or Phase 1b has found duplicate blocks or bad blocks in the root inode (usually inode number 2) for the file system.

Possible responses to the CONTINUE prompt are:

YES — Ignore the **DUPS/BAD** error condition in the root inode and attempt to continue to run the file system check. If the root inode is not correct, then this may result in a large number of other error conditions.

NO — Terminate **fsck**.

I OUT OF RANGE I=I NAME=F (REMOVE)

A directory entry *F* has an inode number *I* which is greater than the end of the inode list.

Possible responses to the REMOVE prompt are:

YES — The directory entry *F* is removed.

NO — Ignore this error condition.

UNALLOCATED I = I OWNER = O MODE = M SIZE = S MTIME = T DIR = F (REMOVE)

A directory entry *F* has a directory inode *I* without allocate mode bits. The owner *O*, mode *M*, size *S*, modify time *T*, and directory name *F* are printed.

Possible responses to the REMOVE prompt are:

YES The directory entry *F* is removed.

NO Ignore this error condition.

UNALLOCATED I = I OWNER = O MODE = M SIZE = S MTIME = T FILE = F (REMOVE)

A directory entry *F* has an inode *I* without allocate mode bits. The owner *O*, mode *M*, size *S*, modify time *T*, and file name *F* are printed.

Possible responses to the REMOVE prompt are:

YES The directory entry *F* is removed.

NO Ignore this error condition.

DUP/BAD I = I OWNER = O MODE = M SIZE = S MTIME = T DIR = F (REMOVE)

Phase 1 or Phase 1b has found duplicate blocks or bad blocks associated with directory entry *F*, directory inode *I*. The owner *O*, mode *M*, size *S*, modify time *T*, and directory name *F* are printed.

Possible responses to the REMOVE prompt are:

YES The directory entry *F* is removed.

NO Ignore this error condition.

DUP/BAD I = I OWNER = O MODE = M SIZE = S MTIME = T FILE = F (REMOVE)

Phase 1 or Phase 1b has found duplicate blocks or bad blocks associated with directory entry *F*, inode *I*. The owner *O*, mode *M*, size *S*, modify time *T*, and file name *F* are printed.

Possible responses to the REMOVE prompt are:

YES The directory entry *F* is removed.

NO Ignore this error condition.

ZERO LENGTH DIRECTORY I = I OWNER = O MODE = M SIZE = S MTIME = T DIR = F (REMOVE)

A directory entry *F* has a size *S* that is zero. The owner *O*, mode *M*, size *S*, modify time *T*, and directory name *F* are printed.

Possible responses to the REMOVE prompt are:

YES The directory entry *F* is removed; this will always invoke the **BAD/DUP** error condition in Phase 4.

NO Ignore this error condition.

DIRECTORY TOO SHORT I = I OWNER = O MODE = M SIZE = S MTIME = T DIR = F (FIX)

A directory *F* has been found whose size *S* is less than the minimum size directory. The owner *O*, mode *M*, size *S*, modify time *T*, and directory name *F* are printed.

Possible responses to the FIX prompt are:

YES Increase the size of the directory to the minimum directory size.

NO Ignore this directory.

DIRECTORY CORRUPTED I = I OWNER = O MODE = M SIZE = S MTIME = T DIR = F (SALVAGE)

A directory with an inconsistent internal state has been found.

Possible responses to the FIX prompt are:

YES Throw away all entries up to the next 512-byte boundary. This rather drastic action can throw away up to 42 entries, and should be taken only after other recovery efforts have failed.

NO Skip up to the next 512-byte boundary and resume reading, but do not modify the directory.

BAD INODE NUMBER FOR '.' I = I OWNER = O MODE = M SIZE = S MTIME = T DIR = F (FIX)

A directory *I* has been found whose inode number for '.' does does not equal *I*.

Possible responses to the FIX prompt are:

YES Change the inode number for '.' to be equal to *I*.

NO Leave the inode number for '.' unchanged.

MISSING '.' I = I OWNER = O MODE = M SIZE = S MTIME = T DIR = F (FIX)
A directory *I* has been found whose first entry is unallocated.

Possible responses to the FIX prompt are:

YES Make an entry for '.' with inode number equal to *I*.

NO Leave the directory unchanged.

MISSING '.' I = I OWNER = O MODE = M SIZE = S MTIME = T DIR = F

CANNOT FIX, FIRST ENTRY IN DIRECTORY CONTAINS F
A directory *I* has been found whose first entry is *F*. **Fsck** cannot resolve this problem. The file system should be mounted and the offending entry *F* moved elsewhere. The file system should then be unmounted and **fsck** should be run again.

MISSING '.' I = I OWNER = O MODE = M SIZE = S MTIME = T DIR = F

CANNOT FIX, INSUFFICIENT SPACE TO ADD '.'
A directory *I* has been found whose first entry is not '.'. **Fsck** cannot resolve this problem. This problem should never occur. Contact your Tektronix Field Office.

EXTRA '.' ENTRY I = I OWNER = O MODE = M SIZE = S MTIME = T DIR = F (FIX)
A directory *I* has been found that has more than one entry for '.'.

Possible responses to the FIX prompt are:

YES Remove the extra entry for '.'.

NO Leave the directory unchanged.

BAD INODE NUMBER FOR '..' I = I OWNER = O MODE = M SIZE = S MTIME = T DIR = F (FIX)
A directory *I* has been found whose inode number for '..' does does not equal the parent of *I*.

Possible responses to the FIX prompt are:

YES Change the inode number for '..' to be equal to the parent of *I*.

NO Leave the inode number for '..' unchanged.

MISSING '..' I=I OWNER=O MODE=M SIZE=S MTIME=T DIR=F (FIX)

A directory *I* has been found whose second entry is unallocated.

Possible responses to the FIX prompt are:

YES Make an entry for '..' with inode number equal to the parent of *I*.

NO Leave the directory unchanged.

MISSING '..' I=I OWNER=O MODE=M SIZE=S MTIME=T DIR=F

CANNOT FIX, SECOND ENTRY IN DIRECTORY CONTAINS F

A directory *I* has been found whose second entry is *F*. **Fsck** cannot resolve this problem. Move the offending entry *F* elsewhere, then run **fsck** again.

MISSING '..' I=I OWNER=O MODE=M SIZE=S MTIME=T DIR=F

CANNOT FIX, INSUFFICIENT SPACE TO ADD '..'

A directory *I* has been found whose second entry is not '..'. **Fsck** cannot resolve this problem. This problem should never happen. Contact your Tektronix Field Office.

EXTRA '..' ENTRY I=I OWNER=O MODE=M SIZE=S MTIME=T DIR=F (FIX)

A directory *I* has been found that has more than one entry for '..'.

Possible responses to the FIX prompt are:

YES Remove the extra entry for '..'.

NO Leave the directory unchanged.

Phase 3 — Check Connectivity

UNREF DIR I = I OWNER = O MODE = M SIZE = S MTIME = T (RECONNECT)

The directory inode *I* was not connected to a directory entry when the file system was traversed. The owner *O*, mode *M*, size *S*, and modify time *T* of directory inode *I* are printed.

Possible responses to the RECONNECT prompt are:

YES — Reconnect directory inode *I* to the file system in the directory for lost files (usually *lost+found*). This may invoke the *lost+found* error condition in Phase 3 if there are problems connecting directory inode *I* to *lost+found*. This may also invoke the **CONNECTED** error condition in Phase 3 if the link was successful.

NO — Ignore this error condition. This will always invoke the **UNREF** error condition in Phase 4.

SORRY. NO lost+found DIRECTORY

There is no *lost+found* directory in the root directory of the file system; **fsck** ignores the request to link a directory in *lost+found*. This will always invoke the **UNREF** error condition in Phase 4. Check protection modes of *lost+found*. See **fsck** (8) manual entry for further detail.

SORRY. NO SPACE IN lost+found DIRECTORY

There is no space to add another entry to the *lost+found* directory in the root directory of the file system; **fsck** ignores the request to link a directory in *lost+found*. This will always invoke the **UNREF** error condition in Phase 4. Clean out unnecessary entries in *lost+found* or make *lost+found* larger. See **fsck** (8) manual entry for further detail.

DIR I = I1 CONNECTED. PARENT WAS I = I2

This is an advisory message indicating a directory inode *I1* was successfully connected to the *lost+found* directory. The parent inode *I2* of the directory inode *I1* is replaced by the inode number of the *lost+found* directory.

Phase 4 — Check Reference Counts

UNREF FILE I = I OWNER = O MODE = M SIZE = S MTIME = T (RECONNECT)

Inode *I* was not connected to a directory entry when the file system was traversed. The owner *O*, mode *M*, size *S*, and modify time *T* of inode *I* are printed.

Possible responses to the RECONNECT prompt are:

YES — Reconnect inode *I* to the file system in the directory for lost files (usually *lost+found*). This may invoke the *lost+found* error condition in Phase 4 if there are problems connecting inode *I* to *lost+found*.

NO — Ignore this error condition. This will always invoke the **CLEAR** error condition in Phase 4.

(CLEAR)

The inode mentioned in the immediately previous error condition can not be reconnected.

Possible responses to the CLEAR prompt are:

YES — De-allocate the inode mentioned in the immediately previous error condition by zeroing its contents.

NO — Ignore this error condition.

SORRY. NO lost+found DIRECTORY

There is no *lost+found* directory in the root directory of the file system; **fsck** ignores the request to link a file in *lost+found*. This will always invoke the CLEAR error condition in Phase 4. Check protection modes of *lost+found*.

SORRY. NO SPACE IN lost+found DIRECTORY

There is no space to add another entry to the *lost+found* directory in the root directory of the file system; **fsck** ignores the request to link a file in *lost+found*. This will always invoke the CLEAR error condition in Phase 4. Check size and contents of *lost+found*.

LINK COUNT FILE I=I OWNER=O MODE=M SIZE=S MTIME=T COUNT=X SHOULD BE Y (ADJUST)

The link count for inode *I* which is a file, is *X* but should be *Y*. The owner *O*, mode *M*, size *S*, and modify time *T* are printed.

Possible responses to the ADJUST prompt are:

YES Replace the link count of file inode *I* with *Y*.

NO Ignore this error condition.

LINK COUNT DIR I=I OWNER=O MODE=M SIZE=S MTIME=T COUNT=X SHOULD BE Y (ADJUST)

The link count for inode *I* which is a directory, is *X* but should be *Y*. The owner *O*, mode *M*, size *S*, and modify time *T* of directory inode *I* are printed.

Possible responses to the ADJUST prompt are:

YES Replace the link count of directory inode *I* with *Y*.

NO Ignore this error condition.

LINK COUNT F I=I OWNER=O MODE=M SIZE=S MTIME=T COUNT=X SHOULD BE Y (ADJUST)

The link count for *F* inode *I* is *X* but should be *Y*. The name *F*, owner *O*, mode *M*, size *S*, and modify time *T* are printed.

Possible responses to the ADJUST prompt are:

YES Replace the link count of inode *I* with *Y*.

NO Ignore this error condition.

UNREF FILE I=I OWNER=O MODE=M SIZE=S MTIME=T (CLEAR)

Inode *I* which is a file, was not connected to a directory entry when the file system was traversed. The owner *O*, mode *M*, size *S*, and modify time *T* of inode *I* are printed.

Possible responses to the CLEAR prompt are:

YES De-allocate inode *I* by zeroing its contents.

NO Ignore this error condition.

UNREF DIR I = I OWNER = O MODE = M SIZE = S MTIME = T (CLEAR)
Inode *I* which is a directory, was not connected to a directory entry when the file system was traversed. The owner *O*, mode *M*, size *S*, and modify time *T* of inode *I* are printed.

Possible responses to the CLEAR prompt are:

YES De–allocate inode *I* by zeroing its contents.

NO Ignore this error condition.

BAD/DUP FILE I = I OWNER = O MODE = M SIZE = S MTIME = T (CLEAR)
Phase 1 or Phase 1b have found duplicate blocks or bad blocks associated with file inode *I*. The owner *O*, mode *M*, size *S*, and modify time *T* of inode *I* are printed.

Possible responses to the CLEAR prompt are:

YES De–allocate inode *I* by zeroing its contents.

NO Ignore this error condition.

BAD/DUP DIR I = I OWNER = O MODE = M SIZE = S MTIME = T (CLEAR)
Phase 1 or Phase 1b has found duplicate blocks or bad blocks associated with directory inode *I*. The owner *O*, mode *M*, size *S*, and modify time *T* of inode *I* are printed.

Possible responses to the CLEAR prompt are:

YES De–allocate inode *I* by zeroing its contents.

NO Ignore this error condition.

FREE INODE COUNT WRONG IN SUPERBLK (FIX)
The actual count of the free inodes does not match the count in the super–block of the file system.

Possible responses to the FIX prompt are:

YES Replace the count in the super–block by the actual count.

NO Ignore this error condition.

Phase 5 – Check Cyl groups

CG C: BAD MAGIC NUMBER

The magic number of cylinder group *C* is wrong. This usually indicates that the cylinder group maps have been destroyed. When running manually the cylinder group is marked as needing to be reconstructed.

EXCESSIVE BAD BLKS IN BIT MAPS (CONTINUE)

An inode contains more than a tolerable number (usually 10) of blocks claimed by other inodes or that are out of the legal range for the file system.

Possible responses to the CONTINUE prompt are:

YES Ignore the rest of the free–block maps and continue the execution of **fsck**.

NO Terminate **fsck**.

SUMMARY INFORMATION T BAD

where *T* is one or more of:

(INODE FREE)

(BLOCK OFFSETS)

(FRAG SUMMARIES)

(SUPER BLOCK SUMMARIES)

The indicated summary information was found to be incorrect. This error condition will always invoke the **BAD CYLINDER GROUPS** condition in Phase 5.

X BLK(S) MISSING

X blocks unused by the file system were not found in the free–block maps. This error condition will always invoke the **BAD CYLINDER GROUPS** condition in Phase 5.

BAD CYLINDER GROUPS (SALVAGE)

Phase 5 has found bad blocks in the free–block maps, duplicate blocks in the free–block maps, or blocks missing from the file system.

Possible responses to the SALVAGE prompt are:

YES replace the actual free–block maps with a new free–block maps.

NO Ignore this error condition.

FREE BLK COUNT WRONG IN SUPERBLOCK (FIX)
The actual count of free blocks does not match the count in the super–block of the file system.

Possible responses to the FIX prompt are:

YES Replace the count in the super–block by the actual count.

NO Ignore this error condition.

Phase 6 – Salvage Cylinder Groups

This phase concerns itself with the free–block maps reconstruction. No messages are produced.

Cleanup

V files, W used, X free (Y frags, Z blocks)
This is an advisory message indicating that the file system checked contained *V* files using *W* fragment sized blocks leaving *X* fragment sized blocks free in the file system. The numbers in parenthesis breaks the free count down into *Y* free fragments and *Z* free full sized blocks.

CAUTION

If either of the following two messages appears, reboot the system. If the system is not rebooted immediately, all the work done by **fsck** *will be undone by copies of tables the system keeps in memory.*

******** REBOOT UNIX ********
This is an advisory message indicating that the root file system has been modified by **fsck**.

******** FILE SYSTEM WAS MODIFIED ********
This is an advisory message indicating that the current file system was modified by **fsck**. If this file system is mounted or is the current root file system, **fsck** should be halted and UNIX rebooted.

B

Appendix — Hardware Considerations

Introduction

This appendix contains procedures for the removal and replacement of half-size enhancement boards.

Circuit Board Removal

Use the following procedure to remove an enhancement board from the system cabinet:

1. Log off the system. Turn off the system by pressing the start/stop switch located at the lower right of the front panel. The 6100 uses a "soft" power-down procedure, meaning that the system may not power down immediately after the switch is pressed. Wait for the light on the start/stop switch to go out and the disk drive to cycle down to be sure that the system has stopped.
2. After you are sure that the system is off, pull the power cord from the wall outlet. When doing so, grasp the cord by the end near the outlet to avoid damaging the cord.

WARNING

Potentially lethal voltages are present when power is applied to this unit, and are accessible when the cover is removed. Be certain that you unplug the power cord from the power outlet before continuing this procedure.

3. Turn off any peripherals connected to the workstation.
4. Carefully turn the system cabinet so that the back of the cabinet is accessible. Be careful not to stress any connectors or cables.
5. Grasp the cable management cover at the bottom and lift gently. The cover is hinged at the top and opens up, as shown in Figure B–1.
6. Locate the enhancement board that you want to remove.
7. Disconnect any cables connected to ports on the board if you are removing an interface board. Do so by unscrewing any connecting screws, grasping the cable at the connector, and gently pulling. Lay the cables aside.

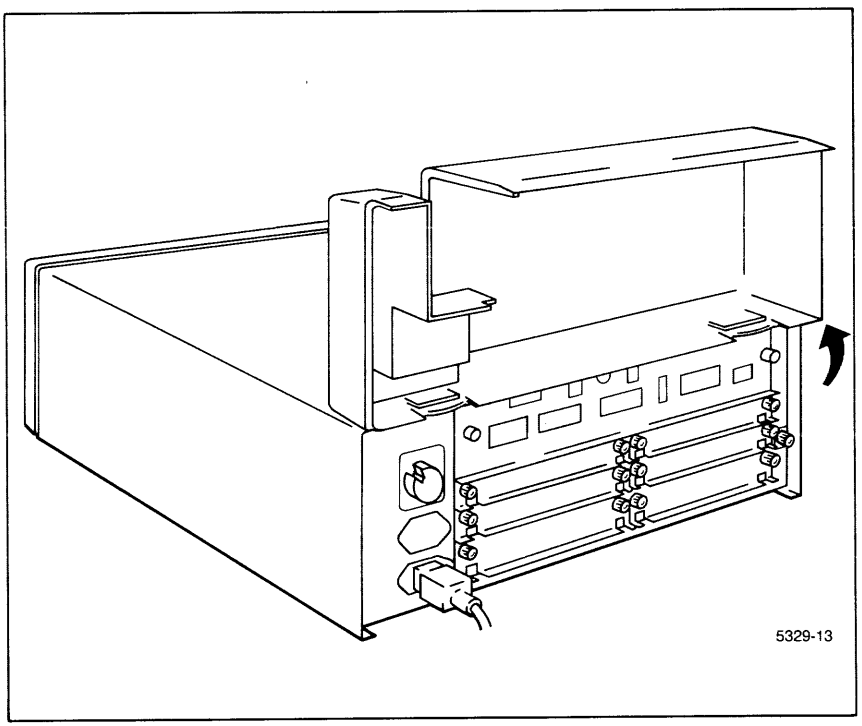


Figure B–1. Opening the Cable Management Cover.

8. There are two mounting thumbscrews (left and right sides) that secure the enhancement board to the system chassis. Loosen both screws until they do not engage the chassis.
9. Remove the circuit board from the chassis by grasping both screws and gently pulling. Pull the board directly out from the cabinet until it is clear of the plastic guides, as shown in Figure B–2. *If you do not expect to install this circuit board or another board before using the system again, be sure to install a cover plate over the vacated slot to permit optimum cooling.*
10. Close the cable management cover.
11. Return the cabinet to its original position.
12. Plug the electrical cord into the wall outlet.

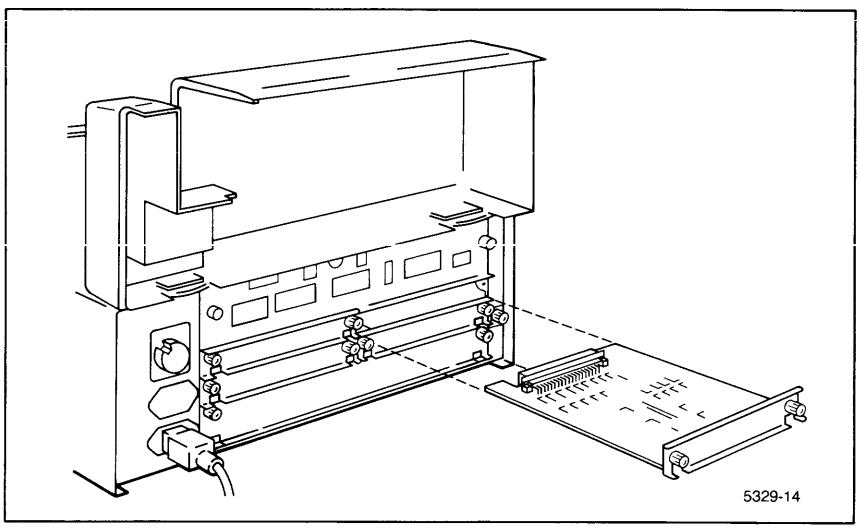


Figure B–2. Removing the Enhancement Board.

Circuit Board Replacement

Use the following procedure to replace the enhancement board.

1. Log off the system. Turn off the system by pressing the start/stop switch located at the lower right of the front panel. The 6100 uses a "soft" power-down procedure, meaning that the system may not power down immediately after the switch is pressed. Wait for the light on the on/off switch to go out and the disk drive to cycle down to be sure that the system has stopped.
2. After you are sure that the system is off, pull the power cord from the wall outlet. When doing so, grasp the cord by the end near the outlet to avoid damaging the cord.

WARNING

Potentially lethal voltages are present when power is applied to this unit, and are accessible when the cover is removed. Be certain that you unplug the power cord from the power outlet before continuing this procedure.

3. Turn off any peripherals connected to the workstation.
4. Carefully turn the system cabinet so that the back of the cabinet is accessible. Be careful not to stress any connectors or cables.
5. Grasp the cable management cover at the bottom and lift gently. The cover is hinged at the top and opens up, as shown in Figure B-1.
6. If there is a cover plate covering the desired slot, loosen the two mounting screws that secure the plate to the chassis and remove the plate. Save the plate for later use.
7. Position the board in front of the appropriate slot so that the components are on the top side of the board and the backplane connector located on the board is facing away from you, as shown in Figure B-2. Any I/O connectors on the enhancement board should be facing towards you. Position the edges of the circuit board along the plastic guides and gently slide the circuit board into the cabinet until the circuit board just touches the backplane connector.

8. Be sure that the connectors are aligned properly. Gently mate the backplane connector by pressing the circuit board until the board connector is fully seated. Do not use excessive pressure. If significant resistance is encountered, check to be sure that the connectors align properly.
9. Once the circuit board is fully seated against the backplane, secure the board by tightening the mounting screws on the faceplate.
10. Attach any peripheral interface cables to the board's connectors. The cable connectors are polarized and only fit one way. Be sure to route the cables so that they are not bent severly. The cable management cover minimizes this possibility, but care should be taken so that no stress is placed on the cable that may cause damage to the cable, connectors, or circuit board.
11. Fold the cable management cover down over the back of the cabinet.
12. Return the system cabinet to its original position.
13. Plug the electrical cord into the outlet.

Software Upgrade Procedures

This section tells how to install a software upgrade to the UTek operating system on your workstation. Upgrading software (and/or system backup) is faster from cartridge tape. This procedure describes both tape and diskette. We recommend you use a cartridge tape. If you do not have a 61TC01 or 4944 contact your Tektronix Field Office,

This section will show you how to:

- Back up your current hard disk
- Install the new kernel
- Install new versions of software
- Restore files from an earlier backup
- Back up your new hard disk

SYSTEM BACKUP

You should back up all software on your system on a regular basis, including the UTek kernel, enhancements, optional software, and data files.

If something happens to the information on the hard disk, such as a system crash, you can then restore everything from the most recent backup, using the *restore* command. For more information about backing up and restoring the system, see *Section 4* of the *System Administration* manual.

Before you install your new version of UTek backup your hard disk to prevent the loss of any special software or usr files.

1. Log in as *root* and type:

   /etc/sysadmin

   and enter the sysadmin password when prompted.

2. The sysadmin interface will display the top level menu.

```
                                Version 2.4

                        System Administration

1: (S)ystem Configuration Maintenance
2: (F)ile System Backup/Restore
3: (I)nstallation of Optional Software
4: (U)ser Login Account Maintenance
5: (G)roup Account Maintenance

Enter ? for general help -->
(Q)uit, Esc Back, (?,H)elp
```

Figure C-1. System Administration Menu

NOTE

If the version number of the System Administration Menu is higher than the one shown above, the menus you see may be slightly different.

Enter 2 to select (F)ile System backup/Restore

For online help, press ?, followed by one of the option numbers (1-5). An explanation of the selected option displays.

The next menu is displayed.

```
            Backup/Restore

1: (L)ist File Systems that have been Dumped
2: (B)backup to Local Media
3: (R)restore from Local Media
4: Backup over Network
5: Restore over Network

Enter ? for general help -->
(Q)uit, Esc Back, (?,H)elp
```

Figure C-2. Backup/Restore Menu

Enter 2 to select (B)backup to Local Media

After typing 2 the sysadmin interface will show you the defaults for the dump. Answer the question for the appropriate device by typing the number opposite the device.

If you have selected floppy diskette, be sure to have enough formatted diskettes to complete the dump. This procedure will use up a large amount of floppies. The sysadmin interface will give you an approximate amount needed to complete the dump. We do not recommend using floppy media for this procedure. Users should use cartridge tape.

Type 0 for the level of the dump. Level 0 meaning that the *entire contents* of your hard disk will be backed up.

Your responses to the questions are displayed and you are asked if you wish to continue. Type <RETURN>.

You can abort the dump any time by typing CTRL C and answering Yes to the question, do you want to abort the dump?

After you have completed the dump of your system hard disk, leave the sysadmin interface and continue with the installation procedure.

OVERVIEW

This overview summarizes the installation steps; each step is presented in detail on the following pages. At this point you should have done a level 0 dump of your hard disk.

1. Install the miniroot file system.
2. Install UTek using the miniroot. The installation is different for different types of source media (diskette and streaming cartridge tape).
3. Remove the software source and store it in a safe place.
4. Boot the system to verify the installation.
5. Restore any user data files.
6. Reinstall optional software packages.
7. Back up the system.

1. INSTALL THE MINIROOT FILE SYSTEM

The miniroot file system serves as a base for creating the real root file system. The following procedure details the steps involved in installing the miniroot system.

The miniroot file system must be loaded before any other software can be loaded.

1. Turn off the workstation.
2. Insert the *standalone utilities* diskette into the drive. The standalone utilities diskette is distributed as a standard accessory with the workstation.
3. Set configuration switch 5 to down and switch 6 to up, specifying "install from diskette," then set configuration switch 4 to down. This lets you select the proper boot file on the diskette. Your setting should look like Figure C-4.

   Figure C-3 shows the location of the switches on the back of the workstation. Figure C-4 shows a detail of the switches in the appropriate positions.
4. Turn on the system console.
5. Turn on the workstation. Press the START/STOP switch and wait until the >>>> prompt appears on the system console. This may take one or two minutes.
6. Enter:

   df(0,0)/sacopy

   Do not enter a space between the right parenthesis) and /**sacopy**. Press <RETURN>.

   The program **sacopy** copies the miniroot file system from diskettes.
7. When the system displays this prompt:

   ```
   Make sure that the first volume of the
   mini root is ready. Press <RETURN> to continue:
   ```

   remove the standalone utilities diskette and store it in a safe place. Insert the miniroot file system diskette, 1 of 4. Then press <RETURN>.

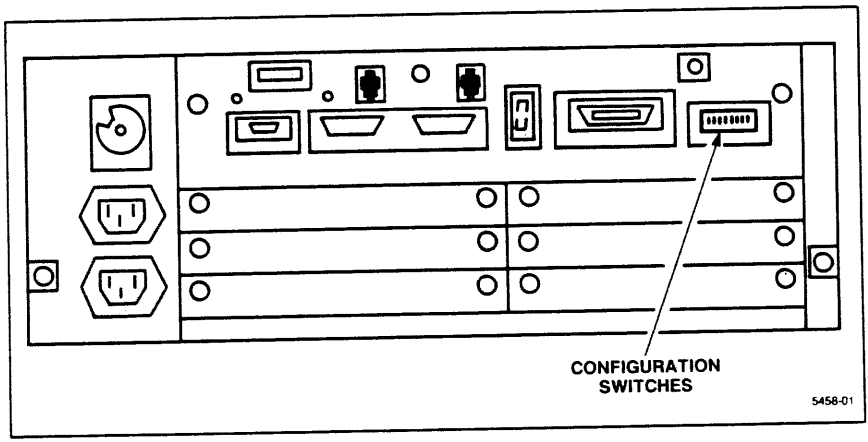


Figure C–3. Back Panel of a Workstation.

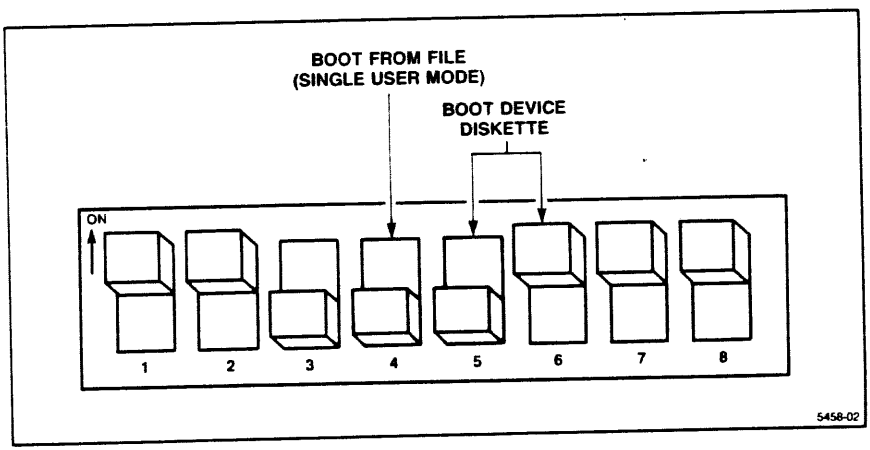


Figure C–4. Configuration Switch Settings for Diskettes.

8. The system asks where you are copying from and to:

```
From: (df(Ø,Ø) ):
To: (dw(1,Ø) ):
```

Press <RETURN> for the defaults at the **From:** and **To:** prompts. This copies from the diskette to the hard disk.

The system begins loading the miniroot into the hard disk's swap space.

9. Enter the default block size and the number of blocks to copy by pressing <RETURN> in response to each question.

When you press <RETURN> in answer to the following questions, the block size defaults to 10240 and the number of blocks defaults to 108. Then press <RETURN>.

```
Block size (1Ø24Ø bytes):
Block size = 1Ø24Ø bytes.
Blocks to copy (default 1Ø8):
Copying 1Ø8 blocks.
```

The system prints a dot as it copies each of the 36 blocks on this diskette.

10. When all the files have been copied from the first diskette, this message appears:

```
Read 36 blocks from 1 volume(s).
Insert next volume.  Press RETURN to continue:
```

Remove the first miniroot diskette and insert the second miniroot file system diskette (2 of 4). Press <RETURN> to continue copying.

The system prints 36 more dots as it copies the blocks on this diskette.

11. Repeat all of step 10 for diskettes (3 o f 4).

12. When the files have been copied from the second diskette, this message appears:

```
Read 72 blocks from 2 volume(s).
Insert next volume.  Press RETURN to continue:
```

Remove the second miniroot diskette and insert the third miniroot file system diskette (3 of 4). Press <RETURN> to finish copying.

13. The system prints 36 more dots and displays this message when it is finished:

```
Copy completed: 1Ø8 blocks copied
Make sure system diskette is in drive.
Press RETURN to continue:
```

Remove the third miniroot diskette. Since the miniroot lets you bring the system up as *root* without a password, store these diskettes in a secure place.

14. Insert the miniroot system diskette, and press <RETURN>.

The root prompt # appears. This indicates that UTek is running in superuser mode, and that you are logged in as *root* on the miniroot. Store these diskettes in a secure place.

2. INSTALL UTEK FROM THE MINIROOT

At this point, the system is ready to load UTek. The procedure you use differs, depending if you have diskettes or a streaming cartridge tape as your source media. Use the appropriate procedure to install UTek once you have the miniroot installed.

Installing UTek from Diskettes

NOTE

Normally you would not do this, but there is a **-r** *option you can use the after* **install**. *This option entirely erases the contents of the hard disk and creates a new file structure. Be aware that using the* **-r** *option also erases any files the user may have. Using* **install** *without the* **-r** *option saves many customized files so you do not have to install them again.* **Install -r** *should be used in the event the update procedure failed.*

1. System releases prior to 2.3 need to save a copy of the syslog.conf prior to installing UTek 2.3.1. Use the command:
   cp /etc/syslog.conf /etc/syslog.conf.old
2. To load UTek, enter

   instaii

   By entering **install** without any options, these files are saved and reinstalled after reinstallation of the core is complete (providing you do not decide to create a new root file system in the next step):

```
.cshrc                          .login                      .profile
diags/guide/menu                etc/assign.classes          etc/daemontab
etc/dumpdates                   etc/exports                 etc/fstab
etc/gidtab                      etc/group                   etc/host.dfs.access
etc/hosts                       etc/hosts.equiv             etc/motd
etc/networks                    etc/passwd                  etc/phones
etc/qconf                       etc/remote                  etc/services
ctc/syslog.conf                 etc/ttys                    etc/ttytype
etc/uidtab                      etc/utmp                    usr/adm/devicelog
usr/adm/lastlog                 usr/adm/wtmp                usr/lib/aliases
usr/lib/aliases.dir             usr/lib/aliases.pag         usr/lib/sendmail.cf
usr/lib/sendmail.fc             usr/lib/crontab             usr/lib/dfs/access
usr/lib/dfs/access.dir          usr/lib/dfs/access.pag      usr/lib/mdqs/log
usr/lib/mdqs/lphdr              usr/lib/uucp/L-devices      usr/lib/uucp/L-dialcodes
usr/lib/uucp/L.cmds             usr/lib/uucp/L.sys          usr/lib/uucp/L.stat
usr/lib/uucp/L_sub              usr/lib/uucp/USERFILE       usr/lib/uucp/local.edit
usr/lib/uucp/uucp.daily         usr/lib/uucp/uucp.hourly    usr/lib/uucp/uucpsummary
usr/lib/uucp/uucpsummary.monthly usr/spool/mail/root        usr/spool/mail/sysadmin
usr/spool/mail/test             etc/gettytab
```

The program asks you to verify that your Distribution media (UTek core) has been loaded:

```
Please be sure the Distribution diskette volume 1
is in the drive. Press <RETURN> when ready.
```

3. Remove the miniroot system diskette and insert the UTek core diskette numbered 1 of *n* (where *n* is the total number of UTek diskettes). Press <RETURN> to continue.

   After a short wait, the installation program displays the names of the files as it moves them from the software source to the hard disk.

4. The system displays this message when it has finished transferring all the files from the first diskette:

```
No more data on /dev/rdf

To continue this installation, insert the next diskette
in the drive, then press <RETURN>.

To quit press <q> followed by <RETURN>.

To install software from a different file or device, type
the new name, then press <RETURN>.

(See the section on multi-volume archives in cpio(1) for
more information.)

->
```

Remove the first UTek diskette from the diskette drive and insert the second diskette (labeled *2 of n*). Press <RETURN>.

The system begins to copy files from the second diskette onto your hard disk.

5. Each time the **No more data** message appears, replace the diskette in the drive with the next diskette of the UTek core package. This procedure takes approximately 45 minutes.

   When all files have been copied, the system displays a message:

```
23Ø4Ø Blocks
***** SOFTWARE EXTRACTION of 613Ø.core SUCCESSFUL *****

Make boot files ...
installing /mnt/diags/diags_os
mkboot complete

Make standard devices ...

Freezing sendmail configuration file ...
Building aliases ...
<22> DEC 21 15:Ø9:41 sendmail[191]:
3 aliases, longest 12 bytes, 55 bytes total

Check compliance of 613Ø.core ...

***** SOFTWARE INSTALLATION of 613Ø.core SUCCESSFUL *****

Restoring the following files:
/etc/umount /dev/dwØØa

install syncing

***** CORE INSTALLATION COMPLETE *****
```

   It is important that you complete the entire process and install all the diskettes. If you must quit, type **q** followed by <RETURN>.

   The system displays this message:

```
Session terminated by user
```

   Remember, this interruption means that the software installation has *not* been completed. To install the software at a later time, you must begin the installation again.

If one of these messages appears:

```
***** CORE EXTRACTION FAILED *****

***** CORE INSTALLATION FAILED *****
```

the installation has not been completed properly. See System Messages for more information.

Installing UTek from Cartridge Tape

NOTE

The following steps assume that the cartridge tape drive is set up as a device (/dev/tc). If it is not, this procedure does not work. See Section 5 for details.

1. System releases prior to 2.3 need to save a copy of the syslog.conf prior to installing UTek 2.3.1. Use the command:
   cp /etc/syslog.conf /etc/syslog.conf.old
2. Insert the UTek cartridge tape in to the drive.
3. You need to create a default catridge tape device using **MAKEDEV**. For example:

   ```
   MAKEDEV /dev/tcn4
   ```

   where *n* is the slot number of the option board.
4. Enter **mt tension 2**. This runs and rewinds the tape twice to ensure there is the correct tension on the tape. This step should always be done if the tape cartridge has not been used recently. If you do not do this step, and the tape is loose, the workstation may make a number of retries before the tape contents are read successfully.
5. Enter **install -t** to load begin loading UTek.

   install -t

6. The system begins the installation process and asks you to verify that your Distribution media (UTek core) has been loaded:

```
Please be sure the Distribution tape is in the drive
(press RETURN when ready):
```

Press <RETURN>. After a short wait, the installation program displays the names of the files as it moves them from the software source to the hard disk.

When all files have been copied, the system displays a message:

```
23Ø4Ø Blocks
***** SOFTWARE EXTRACTION of 613Ø.core SUCCESSFUL *****

Make boot files ...
installing /mnt/diags/diags_os
mkboot complete

Make standard devices ...

Freezing sendmail configuration file ...
Building aliases ...
<22> DEC 21 15:Ø9:41 sendmail[191]:
3 aliases, longest 12 bytes, 55 bytes total

Check compliance of 613Ø.core ...

***** SOFTWARE INSTALLATION of 613Ø.core SUCCESSFUL *****

Restoring the following files:
/etc/umount /dev/dwØØa

install syncing

***** CORE INSTALLATION COMPLETE *****
```

If one of these messages appears:

```
**** CORE EXTRACTION FAILED *****
```

```
***** CORE INSTALLATION FAILED *****
```

the installation has not been completed properly. See System Messages for more information.

3. REMOVE THE SOFTWARE SOURCE

If you have not already done so, remove your software source from its device.

Be sure to handle your software source with care and store it in a safe place. If your hard disk should crash, you might need the source medium to reinstall the software on your system. If you have old copies of UTek stored, replace them with this latest version.

4. VERIFY THE INSTALLATION

1. Type **sync**.
2. Turn off the workstation.
3. Set configuration switches 5 and 6 for autoboot and switch 4 for multiuser mode. (See Figure C-5.)

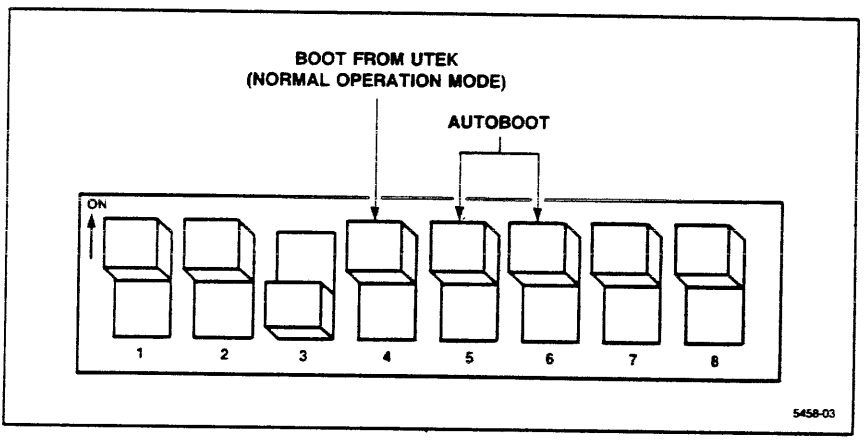


Figure C-5. Configuration Switch Settings for Multi-User Mode

4. Turn the workstation back on. A few power-up messages appear. Then, one of the messages displayed informs you that the system configuration has changed and asks you if you want to update the configuration file. Type **y** in response to this question.

   a. You should answer these questions so the workstation is set the same as it was before reinstalling UTek. For example, if your previous host name was **trolpuppy**, you probably want to name the host **trolpuppy** again. If the distributed file system was previously enabled, you probably want to enable it again.

   b. When you have answered all the configuration questions, the workstation returns a **login:** prompt, verifying that you have successfully installed the UTek core.

   c. You now are running a new version of UTek. Since you used the **install** command without the -r option all of your user accounts should still be intact. At this time, you probably will want to verify this. If any user accounts or data files are missing you can recover them from the backup you made, using the restore command explained later in this appendix.

   When the workstation returns the **login:** prompt, this verifies that you have successfully installed UTek core. However, if you are connected to a LAN, there are three things you probably want to do before using your system:

   a. Log in as root.

   b. Since the */usr/lib/sendmail.cf* file is not saved during the reinstall, you must use the sysadmin interface to reconfigure your system if you are using sendmail. See the discussion in Section 4 on configuring sendmail to find out how to do this.

   c. If your workstation was set up as a file server before the UTek reinstallation, you should go to Section 3 and follow the procedure in that section for setting up the file server. If your workstation was not set up as a file server, continue to the next heading in this procedure.

   If UTek does not work as expected, try installing it again. Go to *Appendix F* for the procedure using **install -r**. If it still doesn't work, contact your Tektronix Field Office for assistance.

5. RESTORING FROM BACKUP

At the beginning of this section you were told to do a backup of your hard disk. This was to prevent the loss of any special software packages or data files. If you need to reinstall any of these any files at this point, proceed with the following procedure.

1. You need to be logged in as root to run the restore command. Type the command *su root*.

   su root

2. Move to the / directory by typing **cd /**.

3. Load the first volume of the backup media into the drive.

   Restore reads tapes dumped with the **dump(8)** command. Its actions are controlled by the *key* argument. The *key* is a string of characters containing at most one function letter and possibly one or more function modifiers. For more information about the *restore* command see the *UTek Command Reference* manual.

4. Type the command restore ivF

   The S flag is for restoring from streaming cartridge tape and the F flag is for restoring from flexible diskette. The i flag is for interactive restore and the v flag is for verbose.

   restore ivF

   The restore command returns you with the prompt `restore>` and gives you dump information and puts you in restore mode. By typing **help** you will get a list of commands and their definitions.

```
Available commands are:
        ls [arg] - list directory
        cd arg - change directory
        pwd - print current directory
        add [arg] - add 'arg' to list of
       files to be extracted
        delete [arg] - delete 'arg' from list
       of files to be extracted
        extract - extract requested files
        quit - immediately exit program
        verbose - toggle verbose flag
       (useful with ''ls'')
        help or '?' - print this list
If no 'arg' is supplied, the current
directory is used
```

5. By typing **pwd** you can see what is on the backup diskette or tape.

```
/
restore > ls
2 *./           23729  backup.errs/    3  lost+found/
2 *../            576  bin/           23  merlin
6  .cshrc       13249  dev/         749Ø  tmp/
8  .files       13825  diags/       3456  usr/
51  .forward       39  dist         9229  usr1/
7  .login          43  ecs         27142  vmunix
5Ø  .mh_profile  1152  etc/
4  backup        288Ø  lib/
```

The number beside the file/directory name is the node number of that file/directory on the media.

6. Type the command **add** to mark a directory or file to be restored. The example shows how paulh is marking the directory

```
restore > add .search
Make node ./paulh
Make node ./paulh/.private
Make node ./paulh/.private/move
Make node ./paulh/.private/move/.search
```

7. Type the command **extract** to request that the files be restored to the hard disk. It returns information about which volume should be installed.

extract

```
Extract requested files
You have not read any dump volumes yet.
Unless you know which volume your file(s)
are on you should start
with the last volume and work towards the first.
Specify next volume #: 2
Mount dump volume 2 then type return
You have read volumes: 2
Specify next volume #: 1
Mount dump volume 1 then type return
extract file ./paulh/.private/move/.search/house
extract file ./paulh/.private/move/.search/queries
Add links
Set directory mode, owner, and times.
set owner/mode for '.'? [yn] y
```

8. After all of the files that you wanted to restore have been restored leave the restore mode by typing **quit**.

6. INSTALL OPTIONAL SOFTWARE

Reinstall any optional software packages or upgrade you software packages, if necessary, to be compatible with the newer Version UTek. These packages came with the *Software Installation* manual, which covers installation of all optional software. Follow the instructions in that installation manual to reinstall each optional software package.

If you have backed up any special files, restore these files at this time. Section 4 of this manual provides you with more information on back up and restore procedures.

7. BACK UP THE SYSTEM

When you have installed and verified your new version of UTek, back up your system. Follow the instructions at the beginning of this section.

SYSTEM MESSAGES

There are two types of system messages you may encounter while installing software:

- UTek messages
- Installation messages

UTek Messages

UTek messages look like this:

```
mv:usr/jenny : no such file or directory (sys2)
```

Each message lists the command that was executing (**mv** in the example above), the message content (**no such file or directory**) and the message tag (**sys2**).

If you need more information on the message, type:

msghlp *message tag*

at the system prompt.

For example, to display information about the example message above, type:

msghlp sys2

Installation Messages

At the end of the installation process, you may see these messages:

```
***** SOFTWARE EXTRACTION FAILED *****
```

The program could not extract the needed files from the source medium — possibly because you terminated the installation process, or inserted diskettes in the wrong sequence, or loaded the source medium incorrectly.

If you are installing software from diskettes, check your diskette drive to be sure you have first loaded the diskette labeled *1 of n* (where n is the total number of diskettes).

Check your source device, to be sure the source medium is correctly loaded, and the device is ready for operation.

When you have checked your source medium and device, run the install procedure again. If you still receive the message, contact your Tektronix Field Office for assistance.

```
***** SOFTWARE INSTALLATION FAILED *****
```

The files as installed do not match the files listed in the Bill of Materials.

The installation program places the results of the compliance check in a file, and displays the name of that file on the screen.

Contact your Tektronix Field Office for assistance.

```
***** CORE INSTALLATION FAILED *****
```

The program could not install the UTek core package on the hard disk. Usually this is caused by a corrupted file system. When you receive this message, you usually also receive one of the following messages:

```
File system was repaired.

File system is corrupted, please repair it.

Unknown error returned from fsck.
```

The *file system was repaired* message indicates there was a problem during installation, but the system was able to repair itself, and it should work correctly.

If you receive the *file system is corrupted, please repair it* message, run fsck in interactive mode on your root file system and try the installation again in *Appendix F*. See Section 7 for details on fsck.

If you receive the *unknown error returned from fsck* message, contact your Tektronix Field Office for assistance.

```
***** SPECIAL DEVICE CREATION FAILED *****
```

The installation program could not create the special devices necessary for UTek.

Contact your Tektronix Field Office for assistance.

```
***** LINK CREATION FAILED *****
```

The installation program could not create the system links as called for by the **/etc/mklinks** program.

Contact your Tektronix Field Office for assistance.

In addition, these error messages can also occur during installation:

```
File system was repaired, continuing with installation.

File system is corrupted, please repair it.

Unknown error returned from fsck.
```

These messages are the result of file system problems while the *fsck* program is running. If the second message occurs, run **fsck** in interactive mode on the root file system (see Section 7). If the last error message occurs, contact your local Tektronix field office.

Cpio Messages

Cpio messages follow the same format as UTek messages: they list the command, the message content, and a message tag, as in this example:

```
cpio: Input not in cpio format (cpio5)
```

For more information on cpio messages, see the *UTek Command Reference*.

List of Terminal Acronyms

INTRODUCTION

This appendix lists the terminal acronyms from the */etc/termcap* file for the most common terminals.

These are the acronyms that you should enter into the Port Configuration form of the sysadmin interface when you are setting up a workstation RS-232-C port for one of these terminals (see Section 4). These are also the acronyms that you should enter if the **TERM =** prompt when you log on doesn't match your terminal type.

Find your terminal type in the righthand column of the following list, and use the corresponding acronym from the lefthand column.

If your terminal type is not in the list, use the **more** command to find it in the */etc/termcap* file:

more /etc/termcap

Or, you can use the **grep** command to search for your terminal type if you can determine a terminal-identifying character string that you think would be in the *termcap* file. See *grep(1)* for details on how to use the **grep** command.

You can also do either of these if your terminal type is in the list, but you want to find out what acronyms are available for special configurations in the *termcap* file.

LIST OF TERMINAL ACRONYMS

TERMCAP acronym	Manufacturer/Model
Tektronix Models	
4012	Tektronix 4012
4014	Tektronix 4014
4025	Tektronix 4025A, 4027A
4105	Tektronix 4105
4107	Tektronix 4107
4115	Tektronix 4115B
4205	Tektronix 4205
4207	Tektronix 4207
4208	Tektronix 4208
8500	Tektronix CT8500
Other Models	
aa	Ann Arbor
aaa	Ann Arbor Ambassador
gigi	DEC Gigi
gt40	DEC GT40
gt42	DEC GT42
vt50	DEC VT50
vt100	DEC VT100
vt102	DEC VT102
vt125	DEC VT125
vt132	DEC VT132
dw1	DecWriter I
dw2	Decwriter II
dw4	Decwriter IV
1620	Diablo 1620, 1640
aj830	Anderson Jacobs 830
5520	NEC Spinwriter 5520
qume5	Qume Sprint 5
1720	Xerox 1720, 1750
dm1520	Data Media 1520
dm2500	Data Media 2500
dm3025a	Data Media 3025
dm3045	Data Media 3045
h1000	Hazeltine 1000

TERMCAP acronym	Manufacturer/Model
h1420	Hazeltine 1420
h1500	Hazeltine 1500
h1510	Hazeltine 1510
h1520	Hazeltine 1520
h1552	Hazeltine 1552
h2000	Hazeltine 2000
esprit	Hazeltine Esprit
8001	Compucolor 8001
3101	IBM 3101
trs80	Tandy TRS80
cit80	C. Itoh CIT 80
cit101	C. Itoh CIT 101
dg6053	Data General 6053
sun	SUN Microsystems
vi200	Visual 200
regent100	ADDS
regent20	ADDS
regent25	ADDS
regent40	ADDS
regent60	ADDS
regent980	ADDS
c100	Concept 100
c108	Concept 108
2621	Hewlett-Packard 2621
2626	Hewlett-Packard 2626
2645	Hewlett-Packard 2645
2648	Hewlett-Packard 2648
hp	Hewlett-Packard
h19	Heath 19
adm2	ADM 2
adm3	ADM 3
adm5	ADM 5
adn31	ADM 31
adm42	ADM 42
pe550	Perkin-Elmer 550
fox	Perkin-Elmer 1100
owl	Perkin-Elmer 1200
ti745	Texas Instrument Silent 700
ti800	Texas Instrument Omni
tvi912	Televideo 912
tvi920	Televideo 920
tvi925	Televideo 925
tvi950	Televideo 950

INDEX

MANUAL REVISION STATUS

PRODUCT: 6130 Intelligent Graphics Workstation

This manual supports the following versions of this product: **V 3.0**

REV DATE	DESCRIPTION
DEC 1984	Original Issue
JAN 1985	Revised: pages 1-3, 1-6, 2-9, 4-32, 4-33
APR 1985	Revised to support UTek Version **2.1**
OCT 1985	Revised to support UTek Version **2.2** Part number rolled to 070-5329-01
DEC 1985	Revised: pages 1-7, 2-6, 2-20, 4-3, 4-4, 4-57, 4-60, 4-63
FEB 1986	Revised: pages 5-23, C-17
APR 1986	Revised: pages 5-23
OCT 1986	Revised to support UTek Version **2.3**
MAY 1987	Revised to support UTek Version **2.3.1**
SEP 1988	Revised to support UTek Version **3.0** and NFS
JAN 1989	Revised: pages v, viii, 9-3, C-5 through C-14 Replaced: 9-12 through the end of section