Operating Systems and Languages Library

XENIX V

Installation and System Administration

User Guide

olivetti

PREFACE

This manual is an installation and system administration guide for the XENIX V operating system. It is for anyone who wishes to install and/or manage this operating system on an Olivetti Personal Computer.

SUMMARY

The first part of the XENIX V Installation and System Administration Guide is devoted to a description of the operating system's installation procedures.

The second part of the guide describes the various activities that are the responsibility of the system administrator or manager. These include starting and stopping the system (Chapter 3), setting up user accounts (Chapter 4) and managing file systems (Chapter 5-7). Chapter 8 describes how to set up the various hardware elements of your Personal Computer so that they function with XENIX, and Chapter 9 how you configure your system using device drivers. Chapter 10 describes how you solve common problems, Chapter 11 how you establish a Micnet network, and Chapter 12 how you build a communication system.

RELATED PUBLICATIONS:

M28 Installation and Operations Guide (Code 4024950 J)

XENIX V User Guide (Code 4022940 Y)

XENIX V User and System Administrator Reference Manual (Code 4022320 Z)

XENIX V System and Application Software Development Tools User Guide (Code 4022990 B)

XENIX V System and Application Software Development Tools Reference Manual (Code 4031710 V)

XENIX V Text Processing User Guide (Code 4022980 C)

XENIX V Text Processing Reference Manual (Code 4031720 W)

C-Language under XENIX V User Guide (Code 4022970 T)

C-Language under XENIX V Reference Manual (Code 4033810 Y)

MS-MACRO ASSEMBLER under XENIX V User Guide (Code 4022950 Z)

MS-MACRO ASSEMBLER under XENIX V Reference Manual (Code 4033800 B)

DISTRIBUTION: General (G)

FIRST EDITION: August 1986

SECOND EDITION: October 1986

Copyright © Microsoft Corporation
1980/1985

Copyright © 1986, by Olivetti
All rights reserved

PUBLICATION ISSUED BY:

Ing. C. Olivetti & C. S.p.A.
Direzione Documentazione
77, Via Jervis - 10015 IVREA (Italy)

TRADEMARK NOTICE

- OLIVETTI is a trademark of Ing. C. Olivetti & C., S.p.A.
- OLITERM is a trademark of Ing. C. Olivetti & C., S.p.A.
- SORTP is a trademark of Ing. C. Olivetti & C., S.p.A.
- ASM-86 is a trademark of Digital Research
- CB-86 is a trademark of Digital Research
- CLEO is a trademark of Phone 1 Inc.
- ETHERNET is a trademark of Xeros Corp.
- GW is a trademark of Microsoft Corp.
- IBM is registered trademark of International Business Machines Corp.
- MICROSOFT is a registered trademark of Microsoft Corp.
- MS is a trademark of Microsoft Corp.
- OMNINET is a trademark of Corvus System Inc.
- p-System is a trademark of Softech Microsystems, Inc.
- PC-DOS is a trademark of International Business Machines Corp.
- PEACHPACK is a trademark of Peachtree Software International Ltd.
- SID-86 is a trademark of Digital Research
- TerminALL is a trademark of TOPIC System, Inc.
- TOPIC is a trademark of TOPIC System, Inc.
- UNIX is a trademark of Bell Laboratories
- XENIX V is a trademark of Microsoft Corporation
- Z80 is a registered trademark of Zilog Inc.
- Z8000 is a registered trademark of Zilog Inc.

CONTENTS

PART 1 INSTALLATION

1. INSTALLATION

ABOUT THIS CHAPTER

This chapter provides an introduction to Part 1 of this manual. This part is concerned with XENIX installation.

CONTENTS

OVERVIEW

This first part of the XENIX Installation and System Administration Guide explains how to install XENIX System V/286 on your Olivetti M28 computer. It also explains how you can upgrade an existing XENIX system to XENIX System V/286 without losing the system files you have edited and the user files you have created.

XENIX SOFTWARE SYSTEMS

The complete XENIX operating system is a set of three software packages:

- The Operating System (BASE package)
- The Software Development System (SOFT package)
- The Text Processing System (TEXT package)

The Operating System, or BASE package, contains the XENIX programs you need to create user accounts, manage file systems and perform system maintenance tasks.

The Software Development System, or SOFT package, contains the XENIX programs you need to create, compile, link, and debug assembly and high-level language programs.

The Text Processing System, or TEXT package, contains the XENIX programs you need to create, edit, and typeset documents.

Each system is installed separately. Installation of the Operating System is required before you can install the Software Development or Text Processing Systems.

WHAT YOU GET

The XENIX software packages are distributed as three appropriately numbered and labelled sets of diskettes. The number of diskettes in your distribution will depend on the combination of packages you have purchased.

In all cases you are also provided with a single Installation diskette. This contains the subset of commands you need to boot up a XENIX system on your computer, prepare XENIX partitions on the hard disk and copy to them the contents of the remaining diskettes.

Make sure you have the right diskettes and that they are arranged in an orderly fashion.

THE DISKETTES CONTAINING THE SOFTWARE PACKAGES SHOULD HAVE WRITE-PROTECT TABS ON THEM, BUT THE SINGLE INSTALLATION DISKETTE SHOULD NOT. (This is because the Installation diskette has to be written to during the installation procedure.)

BEFORE YOU START

To install the Operating System on your computer you need:

- One hard disk with a minimum of 20 megabytes of storage
- One diskette drive with a capacity of 1.2 megabytes

If the hardware components of your system are not already connected up, connect them now. See the M28 Installation and Operations Guide packaged with your system for instructions on how to do this.

Before starting the installation procedure you should also know:

- How to turn the computer and hard disk on and off
- How to insert diskettes in the diskette drive
- How to do a system reset

If you have just assembled your M28 for the first time, or if you are unsure about these procedures, you should review the relevant sections of the M28 Installation and Operations Guide.

During the installation, you will need to use the keyboard to enter information. Always type requested names and numbers exactly as shown and always complete each line by pressing the **CR** key. If you make a typing error, you can delete the character you just typed by pressing the **BKSP** key. You can delete everything you have typed on a line by pressing the **DEL** key.

RUNNING CUSTOMER TEST

Each M28 system should have the CUSTOMER TEST run on it, as soon as it is connected up and before any other operation is attempted. CUSTOMER TEST should also be rerun whenever the configuration of the system is changed -- for instance, by the addition of second hard disk.

CUSTOMER TEST is a series of routines on a single diskette. This diskette is packaged with the M28 Installation and Operations Guide, to which you should refer for more detailed operating instructions.

Be especially sure that the hard disk type is set correctly. Use the SETUP option to verify and if necessary modify this and other settings.

When you have run an error-free CUSTOMER TEST you can begin with the installation proper, in the knowledge that each peripheral is known to the system and in good working order.

INSTALLATION PROCEDURE

The installation procedure for the XENIX operating system has a number of major steps. Read through all the steps carefully before beginning the installation procedure. This will help you to complete your installation without trouble. The steps are:

1. Invoking the boot program on the Installation diskette
2. Running badtrack
3. Booting XENIX from the Installation diskette
4. Initializing the hard disk
5. Starting XENIX from the hard disk
6. Installing the XENIX software packages
7. Creating the super-user password
8. Setting the clock
9. Selecting the console keyboard language
10. Configuring the system
11. Creating the first user account

The order of the first seven steps listed is important, and the success of each step depends on the previous steps, so complete each one correctly before proceeding. If at any point you make a mistake and are not sure what to do, start the installation procedure again.

One or more of the remaining steps may not apply in the case of your particular hardware configuration, or may be completed later. You should read the manual sections carefully at this time to see how they apply to your circumstances.

The following sections tell you how to perform each step. When you have completed the installation, keep this guide and the distribution diskettes in a safe place. You will need them again if you wish to reinstall the system for any reason.

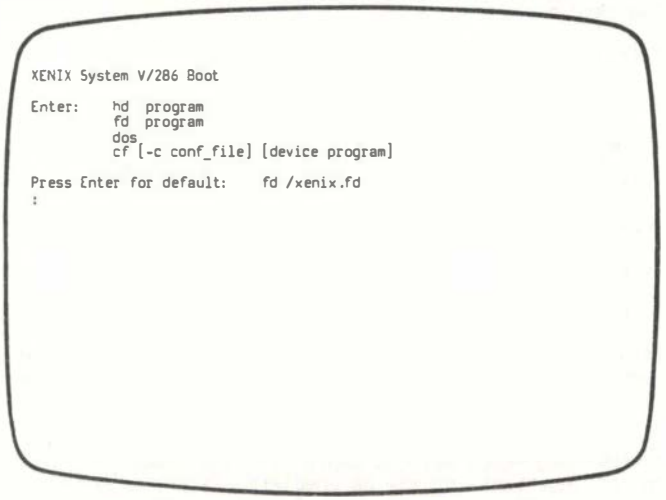
INVOKING THE BOOT PROGRAM ON THE INSTALLATION DISKETTE

The diskette labelled "Installation" contains the subset of XENIX necessary to set up your hard disk XENIX partitions. It also contains the boot program you need to boot up first the **badtrack** program and then the XENIX kernel.

To invoke this boot program:

1. Insert the Installation diskette into the diskette drive and close the drive latch. If there are two diskette drives, select the primary (usually upper) drive. (The diskette drives in XENIX are known as drive 0 and 1, where 0 is the primary, boot drive and 1 the second, optional drive; these equate to the drives known as A and B in MS-DOS and referred to as such in the M28 Installation and Operations Guide.)

2. Switch on the computer and the hard disk(s). If the power is already on, do a system reset by pressing the reset button on your M28. A bootstrap loader loads the boot program from the diskette and the following message appears on your console screen:



Do NOT accept the indicated default.

RUNNING badtrack

You should now run badtrack to ensure that any bad tracks on the hard disk(s) are known to XENIX and therefore avoided in the future.

badtrack only needs to be run once on a properly functioning hard disk. So if you are doing a reinstallation you may skip this phase, unless you are doing so specifically to catch what you believe to be fresh bad tracks.

Note also that if you are adding a second hard disk you must invoke badtrack from the Installation diskette in the manner described here.

1. To invoke badtrack , type:

```
fd /etc/badtrack
```

and press CR .

The message Loading appears on the screen, followed by:

```
Loaded, press ENTER to start
```

2. Press CR again. badtrack begins to scan the primary hard disk (the disk in hard disk drive 0). If badtrack has already been run on this disk, the following message appears:

```
Drive 0 has bad track table already.
Do you want to overwrite ? (y/n)
```

If you get this message and you have no additional bad track information to add, type "n" and press CR . You will then be returned to the boot program prompt, unless you have a second hard disk, in hard disk drive 1, in which case badtrack will move on to scan that.

3. If you type "y" to this message, or if it does not appear, badtrack displays a message such as:

```
drive 0 is type 0
scanning drive 0
cylinder xxx
```

badtrack counts up the cylinders as it checks each one. When it has finished it displays the message:

```
Map for drive 0 is:
```

followed by a listing by cylinder and track of any bad track it has found. It also prompts you with a message such as the following:

```
Enter additional bad tracks for drive 0
Enter cylinder [1-979]
(press Enter to terminate):
```

If you have additional bad track information, enter it now one cylinder and track at a time.

4. As soon as you have no additional information to enter, press CR . badtrack displays an up-to-date bad track map and creates the bad track table.

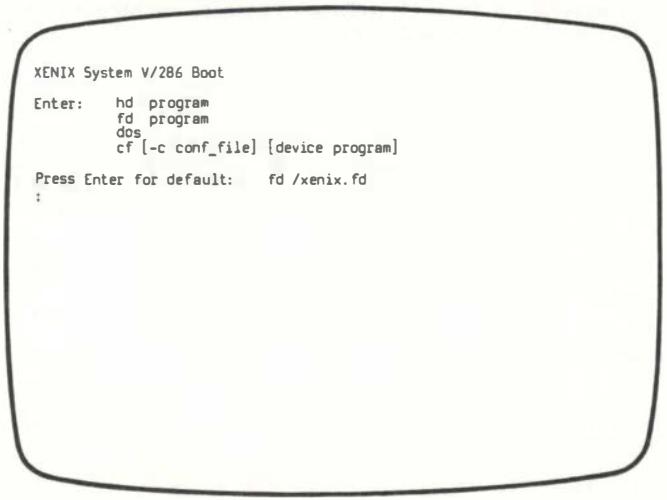
The bad track table is placed in partition 1 on the hard disk. It will be accessed by the disk drive to ensure that no bad tracks are written to.

If you have a hard disk in drive 1, the whole badtrack procedure is now repeated. A second bad track table will be created in partition 1 of that disk.

Finally the boot program prompt returns to the screen. This indicates that the hard disk(s) are ready to receive the XENIX files from the diskette.

BOOTING XENIX FROM THE INSTALLATION DISKETTE

You should have the following boot prompt back on your screen:



This indicates that the default device is the diskette ("fd" standing for floppy disk) and the default pathname /xenix.fd, which is the XENIX kernel on the diskette.

1. Press CR to accept the default.

   To show that it is loading the file, the boot program displays:

   ```
   fd /xenix.fd
   Loading
   ```

   When it has finished loading the file, it displays:

   ```
   Loaded, press Enter to start
   ```

2. Press CR to proceed with installation. The system displays some information about copyright and size of memory and then enters system maintenance mode. At this point, the following message and system prompt is displayed:

   ```
   No single-user login present
   Entering System Maintenance Mode
   ```

```
Terminal type is ansi
#
```

The XENIX kernel is now loaded into memory and you are in a position to prepare the disk in hard disk drive 0 for the installation of the XENIX system.

INITIALIZING THE HARD DISK

You must initialize the disk in hard disk drive 0 with the hdinit program. This program creates the file systems root and usr in partitions 2 and 3 on the hard disk, then copies the various files from the Installation diskette to them.

hdinit also lets you specify whether you are installing XENIX on your computer system for the first time or you are installing XENIX System V on top of an existing version of XENIX.

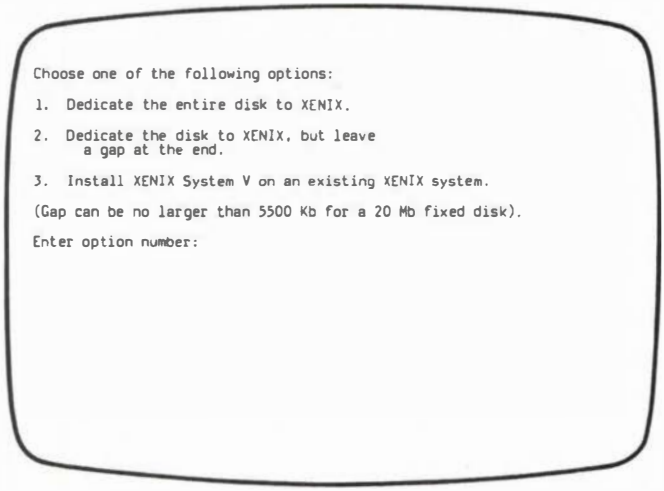
To initialize the hard disk, do the following:

1. Type:

   ```
   hdinit
   ```

   and press CR . hdinit is loaded from the Installation diskette.

2. The following display appears:



   Choose:

- Option 1 if XENIX is going to be the only operating system on your hard disk.

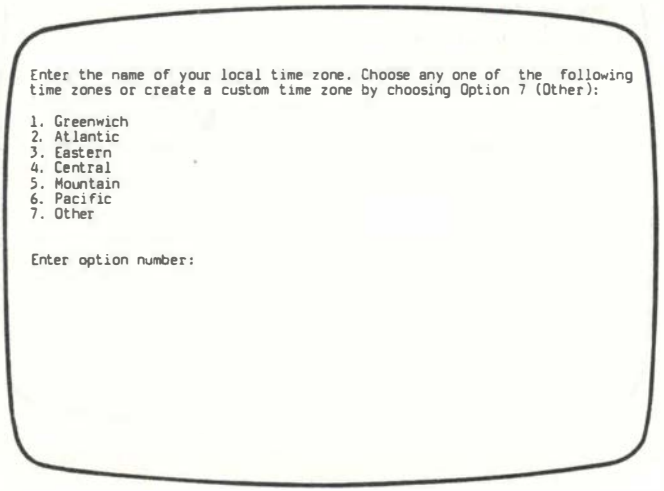
- Option 2 if you want to reserve space for another operating system (such as MS-DOS) in addition to XENIX. If you choose this option, you will be asked how much space you wish to reserve for this other operating system. The valid range of values depends on the size of your hard disk. For a 20 Mbyte hard disk the range is 1 to 5500, which represents the number of Kbyte blocks you wish to reserve. For a 40 Mbyte hard disk the range is 1 to 26562 Kbytes. (As you can see the XENIX partition takes up a minimum of approximately 15 Mbytes.) Type in a valid number and press CR . This creates partition 4 on the hard disk.

- Option 3 if you wish to install XENIX System V on top of an existing XENIX system. If you choose this option, you must skip the rest of the instructions in this section and go to the next section, "Upgrading an Existing XENIX System to XENIX System V". The rest of the instructions in this section apply only if you are installing XENIX on your M28 for the first time.

3. Select the appropriate option, type the corresponding option number, and press CR .

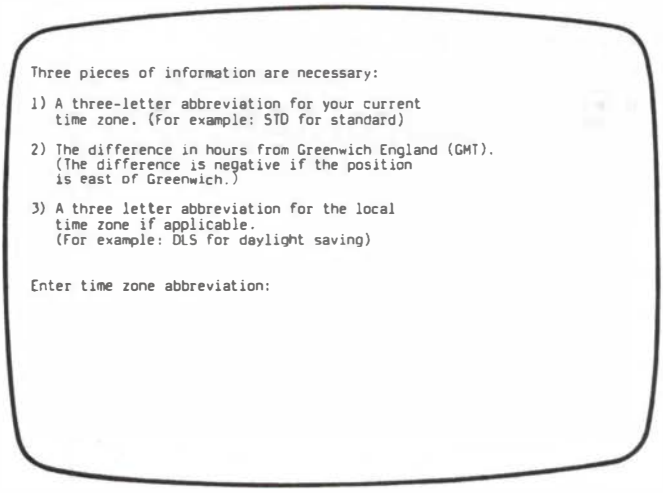
   If you have selected option 3, go the next section. If you have selected option 1 or 2, proceed with these instructions. When you select option 1 or 2, the hdinit program creates the new XENIX partitions..

4. After the disk partitions are created, **hdinit** asks for your time zone. It displays the following message:



5. Select the appropriate time zone, type the corresponding option number, and press CR .

If your particular time zone is not listed, select the "Other" option by typing 7 and pressing CR . In this case the following display appears:



Type a three-letter abbreviation for your particular time zone, then press CR . You are next prompted:

```
Enter hour difference from GMT [-11 to 12]:
```

Type a numeric value representing the number of hours away from Greenwich Mean Time (GMT) you are, then press CR . If you are ahead of GMT the number you type should be negative. For example, if you are in a Western European country such as Italy you should type -1.

You are next prompted:

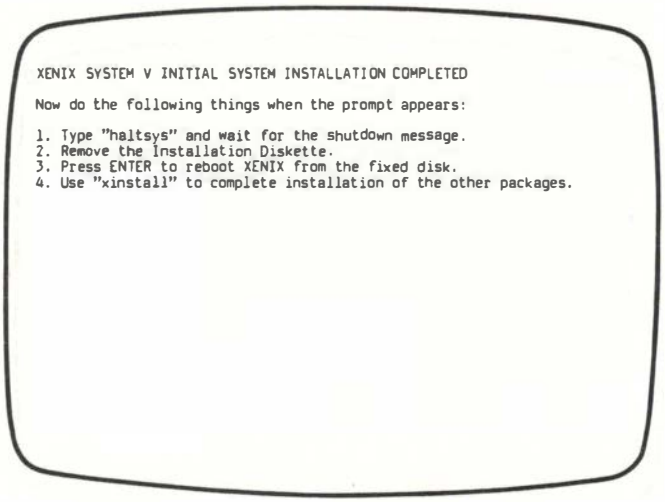
```
Enter daylight time zone abbreviation
(or press ENTER):
```

If you are in a daylight time zone, type a three-letter abbreviation for it and press CR . If this is not applicable, simply press CR .

This concludes the interactive phase of the hdinit program. A lengthy display of messages now appears on your screen, as the various commands required to create the root and usr file systems and copy the contents of the Installation diskette to them are performed

automatically.

When the hdinit program has finished, it displays the following message:



6. Type:

   haltsys

   and press CR &. Wait for the message:

   Normal System Shutdown

   Press ENTER to reboot:

7. Remove the Installation diskette from the diskette drive.

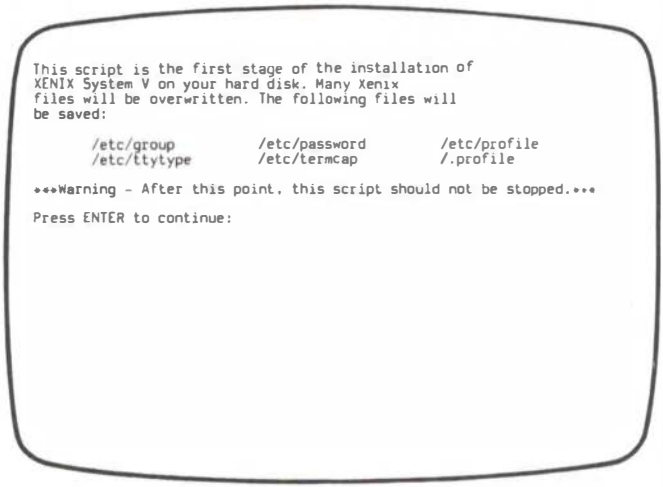
You are now ready to start the XENIX system from the hard disk.

UPGRADING AN EXISTING XENIX SYSTEM TO XENIX SYSTEM V

You may install XENIX System V on a system which is already running a previous version of XENIX. This installation option will save all the files that you may have edited (such as the password file) and will not delete existing user files. It is strongly recommended that you make a complete backup of your system before carrying out this installation option.

To upgrade to XENIX System V:

1. You should be at this step in the installation process as a result of completing Instruction 3 of the previous section, "Initializing the Hard Disk". When you select option 3, the program performs a complete check on the files in the existing XENIX system. Let the program run until you see the message:



The XENIX System V files that correspond to those listed in this message are also installed on the hard disk. However, they are renamed with a ".X5" suffix to keep them from being destroyed by the existing files. For example, the XENIX System V password file will be renamed /etc/passwd.X5.

One file from the existing XENIX system that will be overwritten is the /etc/rc file. If you have modified this file, be sure to save it before continuing with the installation of XENIX System V.

2. Press CR to continue with installation. The hdinit program requires no additional information.

The program prints messages about its progress on the screen. Let the program run until you see the message:

3. Type:

   haltsys

   and press CR . Wait for the message:

   Normal System Shutdown

   Press ENTER to reboot:

4. Remove the Installation diskette from the disk drive.

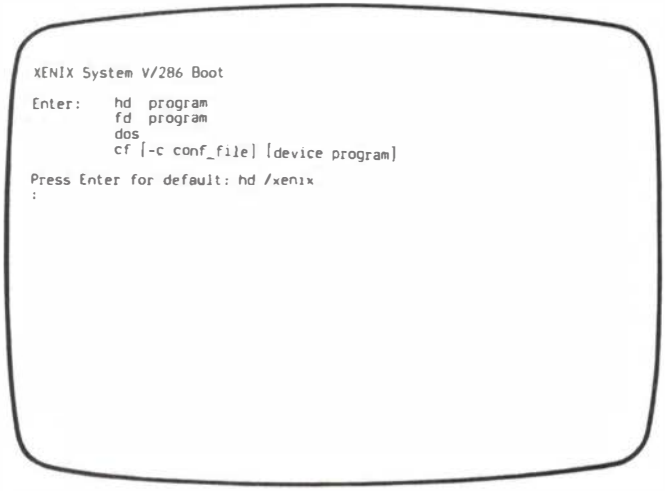
You are now ready to start the XENIX system from the hard disk.

STARTING XENIX FROM THE HARD DISK

You can now start the XENIX system from the newly installed partition on your hard disk. Follow these instructions:

1. Make sure the diskette drive is empty.

2. Press CR . The computer loads the XENIX boot program from the hard disk and displays the following message:



3. Press CR . The program displays the message:

```
hd /xenix
Loading
```

as it loads into memory the XENIX kernel contained in the file xenix on the hard disk.

Next information about copyright and memory size appears on the screen, then the following message and system prompt:

```
No single-user login present
Entering System Maintenance Mode
Terminal type is ansi
#
```

You are now ready to install the XENIX software systems.

INSTALLING THE XENIX SOFTWARE PACKAGES

You can now install the XENIX software systems on your hard disk. There are three standard software systems available: the Base (or Timesharing) System, the Software Development System, and the Text Processing System. The Base System must be installed before the other two systems can be installed. It doesn't matter which of the other two systems is installed next.

To install the software systems use the xinstall command. The three software systems are identified by using the terms base , soft and text as arguments to the xinstall command. So, the command line:

```
xinstall base
```

completes the installation of the Base System, the command line:

```
xinstall soft
```

completes the installation of the Software Development System, and the command line:

```
xinstall text
```

comples the installation of the Text Processing System.

The following set of instructions describes how to install the Base System. Remember that the Base System must always be installed first.

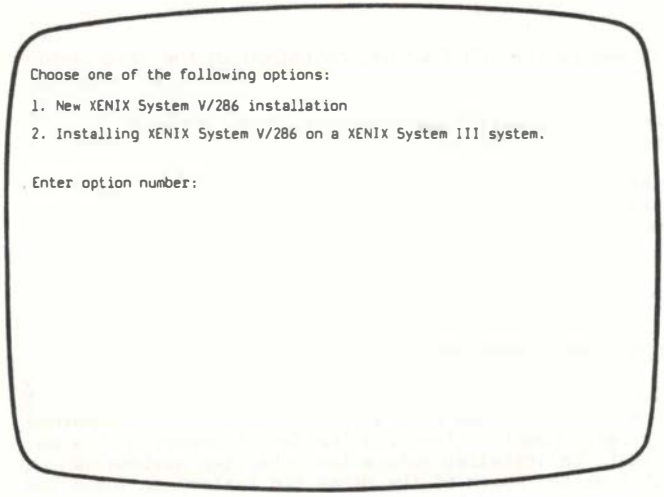
To install the Base System:

1. Examine the diskettes that make up your Base Software System to make sure that all three diskettes are present and in the proper order.

2. Type:

   ```
   xinstall base
   ```

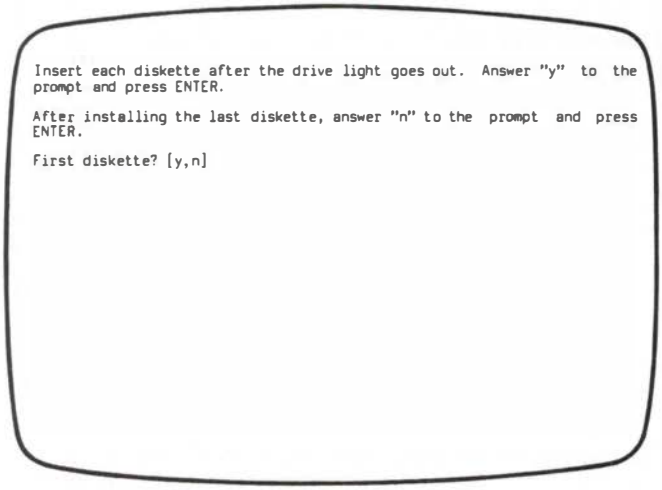
   and press CR . The following message appears:



   Type "2" and press CR if you are updating an existing System III partition. In this case certain files will be preserved rather than overwritten. Otherwise type "1" and press CR

3. The xinstall command first performs a check of the usr file system. It then displays the message:

4. Press **CR** . The following message now appears:



5. Insert the first diskette from the Base System into the diskette drive. If there are two drives, insert the diskette into the upper

drive.

6. Type **y** and press **CR** . The program copies files from the diskette to the hard disk and displays the name of each file as it is copied. When it has finished, it displays the message:

   Next diskette? [y,n]

7. Remove the diskette from the drive and insert the next diskette in the drive.

8. Type **y** and press **CR** . Continue this procedure until all three Base System diskettes have been copied.

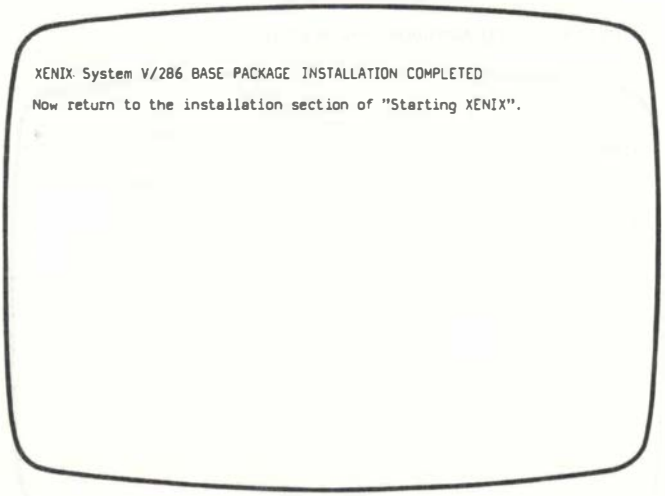
9. When all three Base System diskettes have been copied, type **n** and press **CR** when you see the

   Next diskette? [y,n]

   message. The program assigns permissions for each installed file, instructing you:

   Installation program setting file permissions; please stand by:

   When this has finished, the program displays:



10. Remove the diskette from the diskette drive.

The above procedure must be repeated for each XENIX software package that you want to install on the hard disk.

The procedures for doing a new installation and a reinstallation of an existing XENIX system are identical in terms of the screen displays and responses. The only difference is that the reinstallation option will save modified versions of existing files and will rename their corresponding XENIX System V files with an ".X5" suffix. No files are saved during a reinstallation of the Software Development System or the Text Processing System. The files saved during a reinstallation of the Base System are:

```
/etc/cshrc$                    /usr/lib/mail/faliases
/etc/default/backup            /usr/lib/mail/mailrc
/etc/default/cron              /usr/lib/mail/maliases
/etc/default/dumpdir           /usr/lib/uucp/L.cmds
/etc/default/lpd               /usr/lib/uucp/L-devices
/etc/default/micnet            /usr/lib/uucp/L-dialcodes
/etc/default/mkuser            /usr/lib/uucp/L_stat
/etc/default/passwd            /usr/lib/uucp/L_sub
/etc/default/restor            /usr/lib/uucp/L.sys
/etc/default/su                /usr/lib/uucp/R_stat
/etc/motd                      /usr/lib/uucp/R_sub
/etc/ttys                      /usr/lib/uucp/USERFILE
/usr/lib/mail/aliases.hash
```

When you have finished installing the software systems on your hard disk, you may create the super-user password.

CREATING THE SUPER-USER PASSWORD

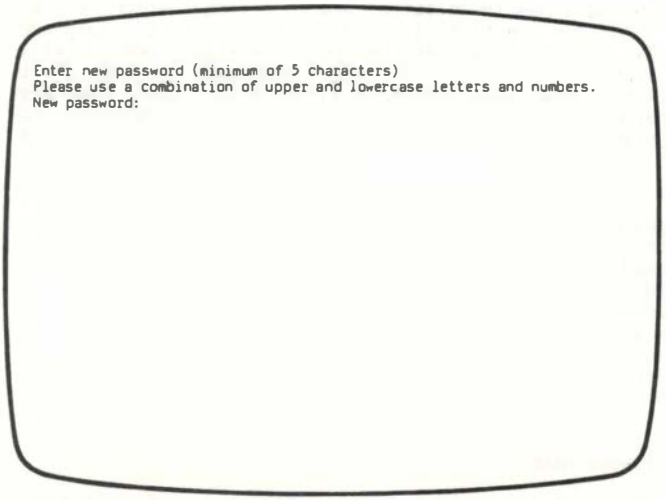
The super-user password keeps the system safe from unauthorized use. It is important that you create a super-user password immediately after the system has been installed to ensure maximum protection of the system and prevent unsafe use of the super-user account. For a complete description of the super-user, see part 2 of this manual.

To create the super-user password:

1. Type:

   ```
   passwd root
   ```

   and press CR . The system displays the message:



The new password can be any sequence of letters, numbers, and/or punctuation marks, but must be at least five characters long.

2. Type the new password and press CR . The system does not display the password as you type it, for security reasons. When you have pressed CR , the system displays the message:

   Re-enter new password:

3. Type the new password once more and press CR . Make sure you type it correctly, otherwise the program will ignore the change.

The super-user password is now in place. From now on, the password will be required whenever you attempt to access the system as the super-user. Do not forget the super-user password. Restoring a forgotten super-user password requires reinstallation of the XENIX system.

SETTING THE CLOCK

You should now stop the system, then start it again in normal operation mode (you are currently in system maintenance mode) and set the system clock correctly.

To stop the system, type

```
haltsys
```

and press CR .

The following message now appears:

```
**Normal System Shutdown**

Press ENTER to reboot:
```

Press CR. The boot program is loaded from the hard disk. You are prompted to identify the operating system kernel you wish to use, and offered the default as follows:

```
Press Enter for default:    hd /xenix
:
```

Press CR again. The XENIX kernel is loaded into memory from the hard disk and the following prompt appears:

```
Type Ctrl-d to continue with normal startup,
(or give root password for system maintenance):
```

Press CTRL D to select normal startup. You may see a number of messages on the screen. The last of these will be the prompt:

```
Console Login:
```

Type the name of the super-user:

```
root
```

and press CR . You are next prompted for the super-user password. Type the password you have just established and press CR . The system opens the super-user account and displays the super-user prompt (#).

At this point you should locate the message on your screen that indicates the date and time. If the message gives the correct date and time all well and good -- you can move on to the next section. If not, you must invoke the setclock command to set the system clock correctly.

To establish the correct date and time type:

```
setclock monthdayhourminuteyear
```

and press CR . The parameter *monthdayhourminuteyear* is a ten-digit figure made up of two-digit values for the month, day, hour, minute and year respectively.

For example, the parameter "1028120086" represents midday on the 10th October, 1986.

When you next start up the system, the date and time should be displayed correctly.

SELECTING THE CONSOLE KEYBOARD LANGUAGE

If your console has a non-U.S. keyboard, you will wish to identify your keyboard language before starting normal operations. Otherwise XENIX will assume you have a U.S. console keyboard and interpret your keyboard input incorrectly.

To select a keyboard language for the console that is other than the U.S. default, use the select command. The syntax of this command is as follows:

```
select countrycode
```

where *country-code* is a two-letter abbreviation for the national version selected. It must be one of the following:

COUNTRY CODE	COUNTRY
DA	Denmark
FS	Finland-Sweden
FR	France
GR	German
IT	Italy
NO	Norway
PO	Portugal
SP	Spain
ES	Spain 2
SF	Swiss-French
SG	Swiss-German
UK	United Kingdom
US	United States

Note that the *country-code* must be entered in upper-case.

Follow the select invocation with the following command:

```
stty -istrip
```

This removes the 7-bit stripping mechanism, allowing the system to interpret keyboard input in the 8-bit character range employed by

national language versions.

For example, to establish the console keyboard as Italian, type:

```
select IT
stty -istrip
```

The keyboard immediately starts behaving as an Italian keyboard.

When you shut down your system, you will lose the select setting you have made. To make a language selection permanent, you should include the appropriate select command in a .login or .profile file.

If your console keyboard has been configured for a language version other than U.S., you can switch back and forth between the default and U.S. versions with simple key combinations.

To switch from a national keyboard to the U.S. version, press:

CTRL ALT F1

To switch back from a U.S. to a default national keyboard version, press:

CTRL ALT F2

CONFIGURING THE SYSTEM

Each combination of hardware elements in a XENIX Personal Computer system must be supported by a subtly different XENIX kernel. The XENIX system which you have just installed relies on a default kernel. This supports a multi-user hardware configuration that includes (in addition to the obligatory console with its hard disk) one or more diskette drives, parallel printers and WS584/VT100 terminals with a U.S. keyboard.

IF YOU HAVE A HARDWARE CONFIGURATION THAT IS OTHER THAN THE DEFAULT -- FOR EXAMPLE, IF YOU HAVE A TAPE DRIVE CONNECTED OR TERMINALS WITH NATIONAL LANGUAGE KEYBOARDS -- YOU MUST RECONFIGURE YOUR XENIX KERNEL ACCORDINGLY. YOU SHOULD DO THIS BEFORE YOU BEGIN NORMAL OPERATIONS.

For full instructions on how to configure your XENIX system, see Chapter 9 of the second part of this guide, "Using Configuration Files".

Whether or not you are reconfiguring your kernel, in all cases where you have a peripheral device such as a terminal, printer or tape drive connected to your M28, you must invoke the relevant XENIX command(s) to set up the connection. See Chapter 8 for instructions on how to add terminals and printers. See Chapter 5 for information on adding a second hard disk, Chapter 7 for information on adding a cartridge tape drive.

CREATING THE FIRST USER ACCOUNT

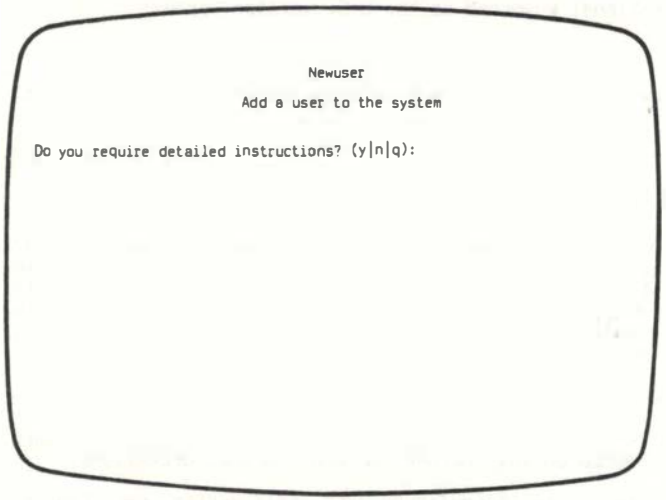
The last step in the installation is to create the system's first user account, "demo". This demonstration account is a temporary workspace on the system that you may use to practice with the XENIX system. Later, after installation is complete and you are familiar with the XENIX commands, you can remove the account and create private accounts for all the users on the system.

To create the first user account:

1. Type:

   mkuser

   and press the **CR** . The system displays the message:



2. Type **n** (for "no") and press **CR** . You can examine the detailed instructions at some other time. The system displays the message:

   Enter new user's login name:

3. Type:

   demo

   and press **CR** . The name "demo" is now the login name for the new user account. Next, the program asks for the user's group:

   Do you want to use the default group? (y|n|q):

4. Type **y** (for yes) and press **CR** . The program then asks for the user's password:

   ```
   Please enter at least 5 characters for the password
   Enter password:
   ```

5. Type:

   ```
   secret
   ```

   or the password of your choice and press **CR** . The program will ask you to retype the password for verification:

   ```
   Re-enter for check:
   ```

6. Retype the password and press **CR** . Next, the program asks you to choose which shell you want to use:

7. Type:

   ```
   1
   ```

   to choose the Bourne shell "sh". Finally, the program asks for comments.

   ```
   Please Enter Comment:   >--------------------
                           >
   ```

8. Type:

   ```
   demo account
   ```

and press CR . The program prompts you as follows:

```
Do you want to change anything? (y|n|q):
```

9. Type n and press CR . The program now displays information about the new account and prompts you as follows:

```
Do you want to add another user? (y|n|q):
```

10. Type n and press CR .

The new demonstration account is now ready. You may use this account to practice logging in, making directories, and running programs.

THE NEXT STEP

This completes the procedure for installing XENIX on the disk in hard disk drive 0. (If you have another hard disk attached to your system, there is a further procedure to go through. This is described in Chapter 5 of the second part of this manual, in the section entitled "Adding a Second Hard Disk".)

If you are familiar with the XENIX operating system, you may continue with normal startup and begin working. Just press CTRL D -- that is, hold down the CTRL key and the D key simultaneously. Refer to the explanation of normal startup in part 2 of this manual if you have problems.

If you are not familiar with the XENIX operating system, we recommend that you halt the system and turn to part 2 of this manual.

To halt the system, follow these steps:

1. Type:

```
/etc/haltsys
```

and press CR . The system displays the following message:

```
** Normal System Shutdown **

Press Enter to reboot:
```

2. Switch off your computer.

Make sure you see the shutdown message before you turn off the power.

PART 2 SYSTEM ADMINISTRATION

2. SYSTEM ADMINISTRATION

ABOUT THIS CHAPTER

This chapter provides an introduction to Part 2 of this manual. This part is concerned with system administration.

CONTENTS

OVERVIEW

The XENIX operating system is a powerful system of programs that helps you perform a full spectrum of tasks, from developing high-level and assembly language programs to creating, editing, and typesetting documents. To keep this powerful system running smoothly, the XENIX system requires careful control of its operation and a regular schedule of maintenance. This guide explains how to operate and maintain the XENIX operating system on your computer, ensuring maximum performance with the least number of system problems.

THE SYSTEM ADMINISTRATOR

Every XENIX system should have at least one person in charge of system maintenance and operation. In this guide, that person is called the system administrator. It is the system administrator's duty to ensure the smooth operation of the system and to perform tasks that require special privileges.

Depending on the size of the system and the number of users on the system, a system administrator's job can be anything from a once-a-week task to a full-time job. Even if the system is small, the system administrator should faithfully perform each required maintenance task, since sloppy maintenance can affect XENIX performance.

All tasks in this guide are presented from the system administrator's point of view, but many can also be accomplished by ordinary users. Since some of the tasks dramatically change the system's operation, we recommend that, whenever possible, the system administrator perform these tasks. This can prevent unwanted or unnecessary changes to the system.

THE SUPER-USER ACCOUNT

The super-user account is a special account for performing system maintenance tasks. It gives the system administrator unusual privileges that ordinary users do not have, such as accessing all files in the system and executing privileged commands. Many of the tasks presented in this guide require that the system administrator be logged in as the super-user. To do this, the system administrator must know the super-user password created during the installation of the XENIX system (see the first part of this guide).

Users who are authorized to act as the super-user, including the system administrator, should log in as the super-user only when it is necessary to perform a system maintenance task. The system administrator should create a personal user account and use it for day-to-day work, reserving the super-user account for system maintenance tasks only, even if the system administrator is the only person using the system.

The number of individuals who are given the super-user password should be kept to a minimum. Misuse of the super-user powers by naive users can result in a loss of data, programs, and even the XENIX system itself.

THE KEYBOARD

Many keys and key combinations have special meanings in the XENIX system. These keys and key combinations have special names that are unique to the XENIX system and may or may not correspond to the keytop labels on your keyboard. To help you find the special keys, the following table shows which keys on your M28 console correspond to XENIX system keys. Where more than one key is specified (for example CTRL C) you should hold down the first key while pressing the second.

XENIX NAME	KEYTOP	ACTION
BREAK	DELETE	Stop current action and return to the shell. This key is also called the INTERRUPT key.
BACKSPACE	BACKSPACE	Deletes the first character to the left of the cursor.
CTRL D	CRTL D	Signals the end of input from the keyboard; also exits current shell.
CTRL H	ERASE	Deletes the first character to the left of the cursor. Also called the ERASE key.
CTRL Q	CTRL Q	Restarts printing after it has been stopped with CTRL S.
CTRL S	CTRL S	Stops printing at the standard output device (does not stop the program).
CTRL U	CTRL U	Deletes all characters on the current line. Also called the KILL.
CTRL\	CTRL\	Quits current command and creates a core file (Recommended for debugging only).
ESCAPE	ESC	Exits the current mode; for example, exits insert mode when in the editor vi.
CR	ENTER	Terminates a command line and initiates an action from the shell.

USING THIS GUIDE

The tasks presented in this second part of the XENIX Installation and System Administration Guide range from simple tasks requiring little knowledge about XENIX, to complex tasks requiring extensive knowledge about XENIX and your computer.

Each chapter explains the tools and knowledge you need to complete the tasks described in that chapter. In some cases, you may need to refer to other manuals, such as the XENIX User Guide.

Chapter 2 introduces the second part of the XENIX Installation and System Administration Guide.

Chapter 3 explains how to start and stop the XENIX system and how to log in as the super-user, the XENIX system's special system administrator account.

Chapter 4 explains how to create accounts for the users who work on your system, how to assign groups, and how to manage user IDs.

Chapter 5 explains how to format hard disk partitions and diskettes, how to create and mount file systems, how to set permissions, and how to keep the system secure.

Chapter 6 explains how to maintain free space on the root file system and other file systems.

Chapter 7 explains how to create backup copies of the root file system and other file systems, and how to format a cartridge tape.

Chapter 8 explains how to add terminals and printers to the system and how to set up your console.

Chapter 9 explains how to use configuration files to install device drivers.

Chapter 10 explains how to solve system problems such as a jammed lineprinter or a forgotten password.

Chapter 11 explains how to create a multiple system mailing network with Micnet.

Chapter 12 explains how to build a communication system.

Appendix A presents a list of the XENIX system special files and explains how to use these files when creating and maintaining file systems.

Appendix B presents a list of commonly used XENIX directories and log files.

Appendix C presents a list of the configuration files provided as part of the XENIX distribution.

3. STARTING AND STOPPING THE SYSTEM

ABOUT THIS CHAPTER

This chapter explains how to start and stop the XENIX system.

CONTENTS

INTRODUCTION

This chapter explains how to start and stop the XENIX system. It also explains how to log in as the super-user.

STARTING THE SYSTEM

Starting a XENIX system requires more than just turning on the power. It requires:

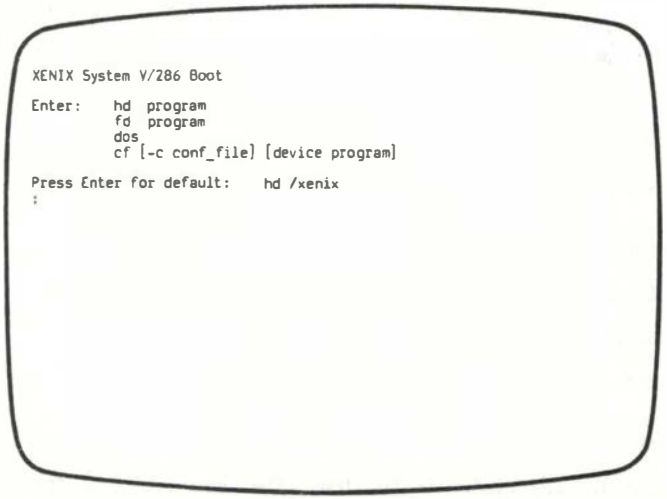
- Loading the operating system
- Cleaning the file system (if the system was not correctly shut down)
- Choosing the mode of system operation

The following sections describe each of these procedures.

LOADING THE OPERATING SYSTEM

The first step in starting the system is to load the operating system from the hard disk. Follow these steps:

1. Make sure that diskette drive 0 (the upper one if you have two) is empty. (Otherwise your M28 will attempt to boot off the diskette.)
2. Switch on the computer and the hard disk(s). (If the power is already on, do a system reset by pressing the reset button on the front of your M28.) A bootstrap loader on the hard disk invokes the XENIX boot program, which displays the message:



3. Press the **CR** key to accept the default value of hd /xenix. The boot program loads the XENIX operating system from the hard disk.

   Note that if you have not responded to the boot prompt by a certain time, the program automatically accepts the default value. This "time-out" feature should be employed with care since it puts you straight into normal operation mode, without giving you the option of entering system maintenance mode. (These two modes are described later in this section.)

When the system is loaded, it displays some information about itself and checks to see if the root file system is clean.

CLEANING THE FILE SYSTEM

The system may not have been shut down properly the last time it was used. In this case, your file systems will need "cleaning". Cleaning involves checking on and repairing damaged files. The XENIX operating system requires clean file systems to perform its tasks.

If you see the message:

```
Proceed with cleaning (y or n)?
```

you should type **y** and press **CR** . The system cleans the root and user file systems, repairing damaged files or deleting files that cannot be repaired. It reports on its progress as each step is completed. At some point, it may ask if you wish to salvage a file. Always answer by typing **y** and pressing **CR** . The option to not salvage a file is available for diagnostic purposes only.

When cleaning is complete, the system usually asks you to choose the mode of operation, but it may also display the message:

```
** Normal System Shutdown **

Press Enter to reboot:
```

If it displays this message, you must reload the system. Do this by pressing CR and repeating the steps given in this subsection.

CHOOSING THE MODE OF SYSTEM OPERATION

You may choose the mode of XENIX operation as soon as you see the message:

```
Type Ctrl-d to continue with normal startup,
(or give root password for system maintenance):
```

The system has two modes: normal operation and system maintenance mode. Normal operation is for ordinary work on the system. This is the mode you should choose to allow multiple users to log in and begin work. System maintenance mode is a specialized mode reserved for work the system administrator must do. It does not allow multiple users or the use of any terminal other than the console.

To choose normal operation, press the CTRL D key sequence. The system displays the date and time and begins to execute commands found in the command file /etc/rc described in Chapter 10, "Solving System Problems". When the commands have finished, the system displays the prompt:

```
Console Login:
```

You may then log in as a normal user, as decribed in the XENIX User Guide, or as the super-user, as described in the following section.

To choose system maintenance mode, type the super-user's password (sometimes called the "root password") and press CR . The system displays the date and time and then the maintenance mode prompt (#). The commands in the /etc/rc file are not executed. (Choose system maintenance mode only if you must do system maintenance work that requires all other users to be off the system.)

LOGGING IN AS THE SUPER-USER

The super-user is the one user account that is created automatically at installation time. Its name is invariably "root", but its password is as the system administrator defined it during the installation.

Many system maintenance tasks, when performed during normal operation, require that you log in as the super-user. For example, you must be logged in as the super-user to stop the system.

Before you can log in as the super-user, you need the super-user password. You also need to see the Console Login: prompt on your console screen. If you do not see this message, press CTRL D until it appears.

To log in as the super-user, follow these steps:

1. When you see the Console Login: message, type the super-user's login name:

   root

   and press CR . The system asks for the super-user's password.

2. Type the super-user's password and press CR . The system does not display the password as you type it, so type each letter carefully.

The system opens the super-user account and displays the message of the day and the super-user prompt (#).

Take reasonable care when you are logged in as the super-user. In particular, you should be very careful when deleting or modifying files or directories. Avoid using wildcard designators in filenames and frequently check your current working directory. Small errors can cause annoying and unwanted changes to the system and user files. Some errors can cause irretrievable damage to a file or the system.

You can leave the super-user account at any time by pressing CTRL D .

STOPPING THE SYSTEM

Stopping the XENIX system takes more than just turning off the computer. You must prepare the system for stopping by using either the shutdown or the haltsys command. The following sections describe each command.

USING THE SHUTDOWN COMMAND

The shutdown command is the normal way to stop the system and should be used whenever the system is in normal operation mode. It warns other users that the system is about to be stopped and gives them an opportunity to finish their work.

To stop the system with the shutdown command, follow these steps:

1. Log in as the super-user (see the previous section). The system opens the super-user account and displays the message of the day and the super-user's prompt.

2. Type:

   shutdown

   and press CR . The system loads the command, which, in turn,

asks for the number of minutes you wish to elapse before the system stops:

```
Minutes till shutdown? (0-15):
```

3. Type any number from 0 to 15 and press CR . The system displays a warning message at each terminal, asking logged-in users to finish their work and log out. As soon as all users are logged out or the specified time has elapsed, the system closes all accounts, displays the message:

```
Broadcast Message from root
XENIX will now terminate.
```

followed by:

```
** Normal System Shutdown **

Press Enter to reboot:
```

and stops.

You may now turn off the computer.

USING THE HALTSYS COMMAND

The haltsys command may be used to halt the system immediately. In general, it should be used only when no other users are on the system or when the system is in system maintenance mode.

To stop the system with the haltsys command, follow these steps:

1. Log in as the super-user (not required when in system maintenance mode). The system opens the super-user account and displays the message of the day and the super-user prompt.

2. Type:

```
haltsys
```

and press CR . The system displays the message:

```
** Normal System Shutdown **

Press Enter to reboot:
```

and stops.

You may now turn off the computer.

4. PREPARING XENIX FOR USERS

ABOUT THIS CHAPTER

This chapter explains how to set up user accounts.

CONTENTS

INTRODUCTION

User accounts help the XENIX system administrator keep track of the people using the system and control their access to the system's resources. Ideally, each user should have a user account. Each account has a unique login name and password with which the user enters the system, and a home directory where the user works.

It is the system administrator's job to create accounts for all users on the system. It is also the administrator's job to maintain user accounts by changing user passwords, login groups, and user IDs when necessary.

This chapter explains how to:

- Add user accounts to the system
- Change an account's password
- Force new passwords
- Create a group
- Change an account's login group
- Change an account's user IDs
- Remove user accounts from the system

The following sections describe each task in detail.

ADDING A USER ACCOUNT

You may add a user account to the system with the mkuser program. The program creates a new entry in the XENIX system's /etc/passwd file. This entry contains information about the new user, such as login name and initial password, that the system uses to let the user log in and begin work. The program also creates a home directory for the user, a mailbox for use with the mail command, and a .profile file, which contains XENIX commands that are executed when the user logs in.

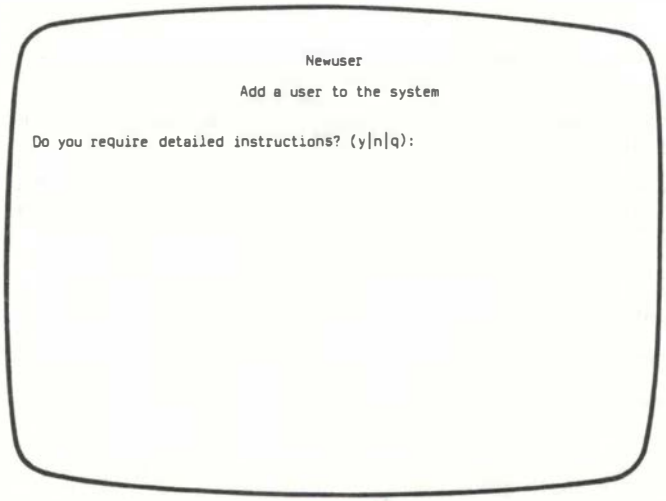
To create a new user account, follow these steps:

Note: At any point you can press DEL to abort the operation and return to the shell prompt.

1. Log in as the super-user.
2. Type:

   mkuser

   and press CR . The system displays the following message:



3. Type the letter **y** and press **CR** if you want full explanatory details on your screen about each element of the program; otherwise, type the letter **n** and press **CR** . (Select **q** for "quit" only if you wish to stop the program and return to the shell prompt.)

   When the program continues, it asks you to enter the new user's login name:

   Enter new user's login name:

   The login name is the name by which XENIX will know the user. It is usually a short version of the user's actual name, typed in lowercase letters. For example, either johnd (a first name and last initial) or jdoe (a first initial and last name) is acceptable for the user John Doe.

   Login names must be a minimum of three and a maximum of eight characters long. They must begin with a lower case character and contain only lower case characters and digits.

4. Type the new name and press **CR** . The program next asks for the group name:

   Do you want to use the default group? (y|n|q):

   The group name is the name of the group of users to which the new user will belong when he logs in. Users in a group have access to a common set of files and directories. If you type **y** and press **CR** , XENIX system's default group "group" (with group ID 50) is used.

If you type n and press CR , a list of existing groups is displayed. You are prompted:

```
Do you want to use one of these groups? (y|n|q):
```

If you type y and press CR you are further prompted:

```
Please give group name or number
```

Type the name or number of the selected group and press CR .

If you type n and press CR you are further prompted:

```
Please enter name for new group:
```

Type the name of your choice and press CR . The name you choose should be at most eight characters long and begin with an alphabetic character.

Next you are asked:

```
Please enter number for new group. Or press ENTER for default
number:
```

Enter the number of your choice and press CR , or simply press CR to accept the next available number. Valid numbers range from 50 to 30000.

5. The program next asks you for the initial password:

```
Please enter at least 5 characters for the password
Enter password:
```

The initial password is the password you assign to the new user. The user will use the initial password to enter the account for the first time. Once in the account, the user should create a new password, one that is hard to guess. (See the XENIX User Guide for more information about changing passwords.)

6. Type the password carefully and press CR . For security reasons what you type is not echoed on the screen. The following prompt appears:

```
Re-enter for check:
```

Retype the chosen password and press CR .

7. Next, the program asks you to choose which shell you want to provide for the user:

```
Please specify the type of shell (command interpreter)
this user requires. You can type 1,2,3 or 4 as follows:

1 Standard (Bourne) Shell.
2 Visual Shell.
3 C Shell.
4 /usr/lib/uucp/uucico.

Enter Shell type (1,2,3 or 4) and press ENTER:
```

8. Select the number of the chosen shell and press CR . You are now prompted to enter a comment:

```
Please Enter Comment:  >--------------------
                       >
```

A comment is information about the new user, such as department name and telephone extension. Although the comment is optional, it is useful if you use the finger command to display information about users. If given, the comment must be no more than 20 characters long, including spaces. It must not contain any colons (:). The example:

```
John Doe, 123
```

shows the recommended form for a comment.

9. Type the comment and press CR . If you do not wish to enter a comment, just press CR .

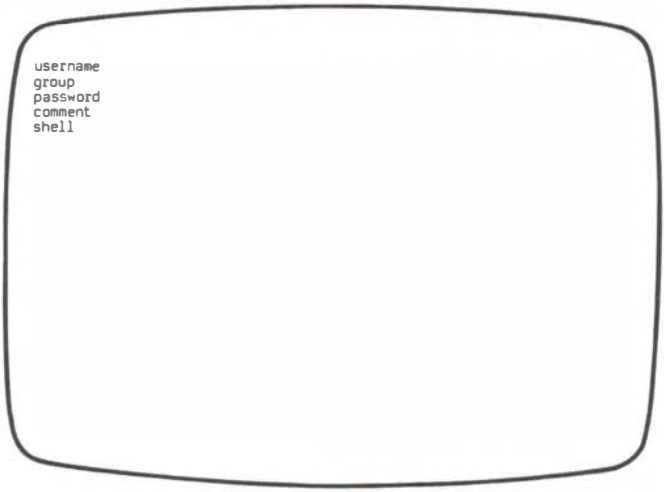
The program now shows what you have typed and the special user entry that it has created for the new user. This entry gets copied to the special system file /etc/passwd. The entry shows the login name, the password (encrypted), the user ID, the group ID, the comment, the user's home directory and the shell. Items in the entry are separated by colons (:).

The program then gives you an opportunity to change the user name, group, password, comment or shell:

```
Do you want to change anything? (y|n|q):
```

10. Type **y** and press **CR** if you wish to change something. Type **n** and skip to step 12 if you wish to complete the new account. (Type **q** for "quit" only if you wish to leave the program and cancel the new account.)

11. If you type **y** in step 10, the program asks for the item you wish to change:



Type the name of the item you wish to change and press **CR** . Then type the new item and press **CR** . The program changes the item and returns to step 10.

12. Once you type **n** in step 10, the program displays the message:

Password file updated

followed by a description of the actions it has taken to add the new user account to the system. The program then asks if you wish to add another user to the system:

Do you wish to add another user? (y|n|q):

13. Type **y** and press **CR** if you wish to add another user. Otherwise, type **n** and press **CR** to stop the program and return to the super-user prompt.

A user can log into a new account as soon as it is created. See the XENIX User Guide for details.

CHANGING A USER'S PASSWORD

Normally, ordinary users can change the passwords of their own accounts with the passwd command (see the XENIX User Guide). Sometimes, however, it may be necessary for the super-user to change the password -- for example, if the user has forgotten the password and cannot get into the account to change it. The super-user may change the password of any user (including the super-user) with the passwd command. To change a password, follow these steps:

1. Log in as the super-user.

2. Type:

   passwd *login-name*

   (where *login-name* is the user's login name) and press CR . The command displays the message:

   Changing password for *login-name*
   Enter new password (minimum of 5 characters)
   Please use a combination of upper and lowercase letters and numbers.
   New password:

3. Type the new password and press CR . The command does not display the password as you type it, so type carefully. The command asks you to type the password again:

   Re-enter new password:

4. Type the password again and press CR .

To see how ordinary users can change their own passwords with the passwd command, see the XENIX User Guide.

FORCING A NEW PASSWORD

From time to time, a user account may need a higher level of security than ordinary. Since the security of any account depends on its password, it is important to keep the password as secret as possible. One way to provide greater security is to force users to change their passwords on a regular basis.

You can force users to change their passwords by using the pwadmin command. This command automatically dates each password and requires the user to provide a new password when a specified number of weeks have passed. The command also requires users to wait a minimum number of weeks before allowing them to restore their previous password. To use the pwadmin command, you must log in as the super-user.

You can enable password aging for a specified user by using the -a option. Use the syntax:

pwadmin -a *login-name*

where *login-name* is the login name of a user.

This activates password aging for the selected user and displays a message such as:

```
Minimum number of weeks: 0.
Maximum number of weeks: 4.
```

This indicates that the user will be required to wait a minimum number of 0 weeks before the password can be changed, and will be forced to change the password after a maximum number of 4 weeks have elapsed. The **-a** option uses the default minimum and maximum values found in the /etc/default/passwd file.

You can choose your own minimum and maximum number of weeks by using the **-min** and **-max** options. For example, a common pair of minimum and maximum values is 2 and 8. To set the minimum and maximum dates, use the syntax:

pwadmin -min *num* **-max** *num login-name*

where *num* is a number in the range 0 to 63, and *login-name* is the login name of the user whose password you are administering. Note that the minimum and maximum cannot both be 0, and that the minimum must not be greater than the maximum.

If you are unsure of the current minimum and maximum values for a password, you can display them by using the syntax:

pwadmin -d *login-name*

This command does not change the current values.

If you wish to force a user to change a password immediately, use the syntax:

pwadmin -f *login-name*

The user is asked on the next login to supply a new password.

When a password no longer requires extra security, you can remove the current minimum and maximum values for the password by using the syntax:

pwadmin -n *login-name*

The system will no longer prompt for changes.

For more information about password aging, see **pwadmin (C)** and **passwd (M)** in the XENIX User and System Administrator Reference Manual.

CREATING A GROUP

A group is a collection of users who share a common set of files and directories. The advantage of groups is that users who have a common interest in certain files and directories can share them without revealing them to others. A new group can be created when you create a new user account with the **mkuser** command (see the section earlier in this chapter on adding user accounts). You can also create new groups by modifying the XENIX system file /etc/group using a XENIX text editor. This is the method described in this section.

To create a new group, you need to choose a group name and a group identification number (group ID). You also need to make a list of the users in the new group. The group name may be any sequence of letters and numbers up to eight characters long, and the group ID may be any number in the range 0 to 65535. Both the group name and ID must be unique, i.e., they must be not be the same as any existing group name or ID.

To create a new group, follow these steps:

1. Log in as the super-user.

2. Display the contents of the /etc/group file by typing:

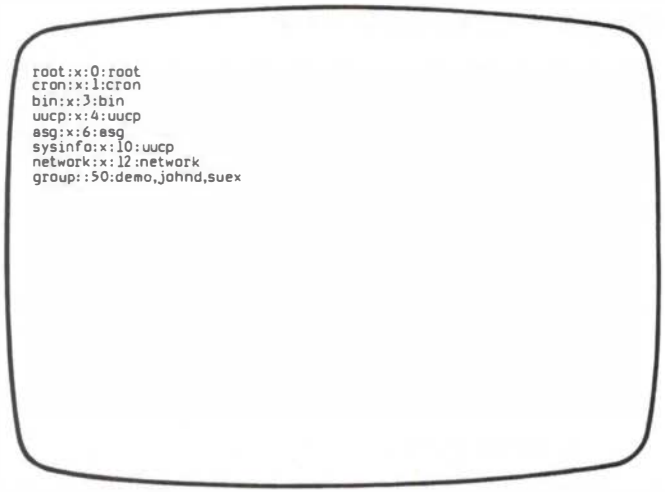
   ```
   cat /etc/group
   ```

   and pressing CR . The **cat** command displays the contents of the /etc/group file. The file contains several user entries, each defining the group name, group ID and users for a group. Each user group entry has the form:

   group-name :: *group-ID* : *users*

   The *users* are shown as a list of login names separated by commas (,).

   For example, a typical /etc/group file might look like this:



In this example only the last entry is a user group entry. The entries with an "x" in them represent system-generated groups. These entries should not be changed in any way.

3. Check the /etc/group file entries to see that the group name and ID you have chosen are unique.

4. If the group name and ID are unique, invoke a XENIX text editor (see the XENIX User Guide) and specify /etc/group as the file to edit.

5. Create a new last line in the file, then insert the new entry in the correct format. For example, if you wish to create a group named "shipping" with group ID "142" and users "johnd", "marym" and "suex", type:

   ```
   shipping::142:johnd,marym,suex
   ```

6. Exit the editor.

To make sure you have entered the group names correctly, use the grpcheck command to check each entry in the /etc/group file. If the new entry is free of errors, no other changes to the file are required.

You can create any number of new groups. Each group may have any number of members. Furthermore, any user may be a member of any number of groups. Multiple group membership is especially convenient for users who have interests that span a variety of areas.

If users are members of several groups, they can gain access to each group by using the newgrp command. See the XENIX User and System Administrator Reference Guide for more information on this command.

CHANGING A USER'S LOGIN GROUP

When users log in, the system automatically places them in their "login group". This is the group given by the group ID in the user's /etc/passwd file entry. You can change a user's login group by changing his group ID. To change the group ID you need the group ID of the new login group, and you need to know how to use a XENIX text editor (see the XENIX User Guide).

To change the group ID, follow these steps:

1. Log in as the super-user.

2. Use the cd command to change the current directory to the /etc directory. Type:

   ```
   cd /etc
   ```

3. Use the cp command to make a copy of the /etc/passwd file. Type:

   ```
   cp passwd passwd+
   ```

4. Invoke a text editor and specify /etc/passwd+ as the file to edit.

5. Locate the desired user's password entry. Each entry begins with the user's login name.

6. Locate the user's group ID number in the user's password entry. It is the fourth item in the entry. Items are separated by colons (:). For example, the entry:

   ```
   marym:9iKlwp:205:50:Mary March, 122:/usr/marym:/bin/sh
   ```

   has group ID "50".

7. Delete the old group ID and insert the new one. Be sure you do not delete any other portion of the user's password entry.

8. Exit the editor.

9. Use the mv command to save the old /etc/passwd file. Type:

   ```
   mv passwd passwd-
   ```

10. Use the mv command to make the edited file the new /etc/password file. Type:

    ```
    mv passwd+ passwd
    ```

You can make sure you have entered the new login group correctly by using the pwcheck command. If the new entry is correct, no other changes to the file are required.

You must not change the group IDs for system accounts such as "cron" and "root". System accounts are any accounts whose user IDs are less than

200. The user ID is the third item in the password entry.

Note that changing a user's login group does not change the "group ownership" of the user's files. Group ownership defines which group has access to a user's files. If users in the new group wish to access the user's files, you must change the group ownership with the chgrp (for "change group") command. For details, see "Changing Group Ownership" in Chapter 5.

CHANGING A USER ID

Sometimes it is necessary to change the user ID in a user's account entry to allow a user to access files and directories transferred from other computers. In particular, if a user has different accounts on different computers and frequently transfers files and directories from one computer to another, then the user IDs in each of the user's account entries must be made the same. You can make them the same by modifying the account entries in the /etc/passwd file.

To change a user ID, follow these steps at every computer for which the user has an account:

1. Log in as the super-user.

2. Use the cd command to change the current directory to the /etc directory. Type:

   ```
   cd /etc
   ```

3. Use the cp command to make a copy of the /etc/passwd file. Type:

   ```
   cp passwd passwd+
   ```

4. Invoke a XENIX text editor (see the XENIX User Guide if in doubt) and specify /etc/passwd+ as the file to edit.

5. Locate the user's account entry. Each entry begins with the user's login name.

6. Locate and substitute the current user ID. The ID is the third item in the entry. For example, the entry:

   ```
   marym:9iKlwp:205:50:Mary March, 122:/usr/marym:/bin/sh
   ```

   has user ID "205".

7. Exit the text editor.

8. Use the mv command to save the old /etc/passwd file. Type:

   ```
   mv passwd passwd-
   ```

9. Use the mv command to make the edited file the new /etc/passwd file. Type:

```
mv passwd+ passwd
```

No other changes to the file are required.

In most cases, you can change the user ID to the same number as the user's most-used account. But the new number must be unique at every system for which the user has an account. If there is any conflict (for example, if the number already belongs to another user on one of the systems), you must choose a new number. You can choose any number greater than 200. Just make sure it is unique, and that you copy it to all systems on which the user has an account.

Once a user's ID has been changed, you must change the "user ownership" of the user's files and directories from the old user ID to the new one. You can do this with the chown (for "change owner") command described in Chapter 5, "Using File Systems". For example, to change the ownership of johnd's home directory, type:

```
chown johnd /usr/johnd
```

This associates the new user ID belonging to johnd with the /usr/johnd directory, in place of the old user ID.

Note that you may use the find command described in Chapter 6, "Maintaining File Systems", to locate all files and directories with the user's old user ID.

REMOVING A USER ACCOUNT

It is sometimes necessary to remove a user account from the system. You can remove a user account with the rmuser program. The program deletes the user's entry from the /etc/passwd file and removes the user's home directory and mailbox.

Before you can remove the user account, you must remove all files and directories from the user's home directory or move them to other directories. If you wish to save the files, you may use the tar command to copy the files to a diskette (see Chapter 7, "Backing Up File Systems").

To remove a user account, follow these steps:

1. Log in as the super-user.

2. Type:

   ```
   cd /usr/ login-name
   ```

   and press CR to change to the user's home directory. The *login-name* must be the user's login name.

3. Make sure that you have made copies of all important files and directories in the user's home directory.

4. Use the **rm** (for "remove") command to remove all files and directories from the user's home directory. This includes any files that begin with a period (.). Directories can be removed by using the **-r** (for "recursive") option of the **rm** command. For example, the command:

   ```
   rm -r bin
   ```

   removes the directory named bin and all files within this directory.

5. After removing all files and directories, make sure the user's mailbox is empty. Type:

   ```
   cat /usr/spool/mail/ login-name
   ```

   and press CR . The *login-name* is the user's login name. If the mailbox contains text, type:

   ```
   cat /dev/null >/usr/spool/mail/ login-name
   ```

   and press CR .

6. When the user's home directory and mailbox are empty, type:

   ```
   cd /usr
   ```

   and press CR . The user's home directory cannot be removed until you have moved to another directory.

7. Type:

   ```
   rmuser
   ```

   and press CR . The program displays a message explaining how to remove a user. This concludes as follows:

   ```
   Press ENTER when you are ready.
   ```

8. Press CR . The program asks for the login name of the user you wish to remove:

   ```
   Enter name of id to be removed.
   ```

9. Type the user's login name. You should now see the message:

   ```
   Removing user login-name from the system. CONFIRM ? (y/n):
   ```

   where *name* is the login name of the user you want to remove.

10. Type **y** to remove the user from the system. Otherwise, type **n** to stop the removal, or **q** (for "quit") to stop the program. If you type **y** the program removes the user's entry from the /etc/passwd file, the user's mailbox, .profile file, and home directory. The program displays the message:

    ```
    User name removed from the system
    ```

where *name* is the login name of the user you have removed.

The program now gives you a chance to remove another user:

```
Do you want to remove another user? (y/n):
```

11. Type **y** to remove another user. Otherwise, type **n** to stop the program.

Note that the rmuser program will refuse to remove an account that has a system name, such as "root", "sys", "sysinfo", "cron", or "uucp", or a system ID (user ID below 200). Also, the program cannot remove a user account if the user's mailbox still has mail in it, or if the user's home directory contains files other than .profile.

5. USING FILE SYSTEMS

ABOUT THIS CHAPTER

This chapter explains how to use file systems.

CONTENTS

INTRODUCTION

This chapter describes one of the most important responsibilities of a system administrator: controlling and recording users' access to the files and directories on the system. It introduces file systems, permissions, system security, and process accounting.

FILE SYSTEMS

A file system is the XENIX system's way of organizing files on mass storage devices such as hard disks and diskettes. A file system consists of directories, files and the information needed to locate and access these items.

Each XENIX system has at least two file systems. These are the root and usr file systems created in partitions 2 and 3 at installation time (see the first part of this guide for more on the installation procedure).

The most fundamental file system is the root file system. The root file system contains the XENIX kernel itself, along with directories containing all the system utilities provided for the use of the system administrator. When you boot up and select system maintenance mode, the root file system is the only file system available to you. You can travel around the directory structure using the cd command. However, you have no access to user accounts and the ls command tells you that the /usr directory is empty. (See the XENIX User Guide for more on moving around and looking at directories.)

The usr file system contains all the user directories in the XENIX system. When you boot up and type CTRL D to select normal startup, the file /etc/rc is invoked. This contains an entry that "mounts" the usr file system onto the /usr directory in the root file system.

When the usr file system is mounted it appears to be part of the root file system itself. You can now access user accounts and users can log in and move around the complete hierarchical directory structure.

A XENIX system may also have other file systems that contain application programs or additional utilities. One reason for using other file systems is to expand the available storage space of the system. Each additional file system adds its free space to the system's total space. New file systems can be specifically created by a user, then mounted onto the system so they can be used.

You can create a file system with the **mkfs** command. This command sets the size and format of the file system and may also copy some files to the new system. You can mount a file system with the **mount** command. When you have mounted the file system, you may access the files and directories in the file system in the same way as files and directories in the root file system. When you are finished with a file system, you can unmount it with the **umount** command.

You can create new file systems on both hard disks and diskettes. A reason for creating new file systems on diskettes is to establish a

collection of application programs and data files that can be easily mounted and used when needed.

Before a disk can receive a new file system it must be prepared properly. The next section explains how to do this.

PREPARING DISKS

You cannot create a file system on a disk until it has first been prepared. With a diskette, this involves formatting with the format program. With a hard disk, it involves creating a partition with the fdisk program.

FORMATTING DISKETTES

You can format diskettes with the format program. Formatted diskettes are required whenever you create a file system. They are also required when you back up a file system with the sysadmin program (see Chapter 7, "Backing up File Systems").

To format a diskette, follow these steps:

1. Insert a diskette into diskette drive 0 (the upper diskette drive if there are two). Make sure there is no read-only tab on the diskette jacket.

   Note that only a console diskette drive should be used. You should not try to run format using a terminal's diskette drive.

2. Type:

   ```
   format /dev/rfd0
   ```

   Press **CR** . The program formats the diskette.

The filename /dev/rfd0 represents the default raw device filename for diskette drive 0. The diskette will be given a format that is compatible with the diskette drive. Assuming that your diskette drive 0 is the standard 1.2 Mbyte model, the diskette will be formatted 15 sector, double-sided. (See Appendix A for a list of the various possible formats associated with the default.)

If you wish to format a low-density diskette, use "rfd048" instead of "rfd0".

If for some reason you have not properly inserted the diskette into the drive, the program displays an error message and stops.

In general, the system administrator should format spare diskettes in advance. Note that formatting removes all data from the diskette, so if you are formatting a diskette that already contains data, make sure that there is nothing you wish to save.

CREATING HARD DISK PARTITIONS

If you wish to create a new file system on a hard disk, you must first create a fresh hard disk partition.

Each connected hard disk can have a maximum of four partitions. Three partitions on the disk in hard disk drive 0 are created and occupied at installation time. The first partition contains the bad track table, the second and third the root and usr file systems respectively. This leaves room for a single additional partition (but only if during the installation you have elected to leave a gap at the end of the hard disk).

If you have an additional hard disk connected, you will have room for up to three further partitions. The first of the four possible partitions will be taken up by the bad track table (if you are adding a second hard disk after system installation time, see the next subsection, "Adding a Second Hard Disk", for more information).

You can create an additional hard disk partition using the fdisk program. Note, however, that for safety reasons you cannot create or delete partitions on a hard disk that has a currently mounted file system on it. In the case of a second hard disk you have only to do an unmount to remove the restriction. With the primary hard disk, however, the root file system, once booted, is always mounted and cannot be unmounted. So to add a XENIX partition you must stop the system, then boot up the Installation diskette and run the fdisk program from there.

Note also that if you wish to create an MS-DOS partition, you should use the equivalent MS-DOS fdisk command.

To invoke fdisk make sure you are logged in as the super-user. Type:

```
/etc/fdisk rawdevicename
```

where *rawdevicename* is either /dev/rhd00 (for the primary hard disk in drive 0) or /dev/rhd10 (for the hard disk in drive 1). Press CR .

At this point you may receive the following message:

```
Warning: Creation and deletion commands not enabled
```

This means that you have a file system mounted on the hard disk you have selected. If you wish to carry on and create a partition, do as suggested above.

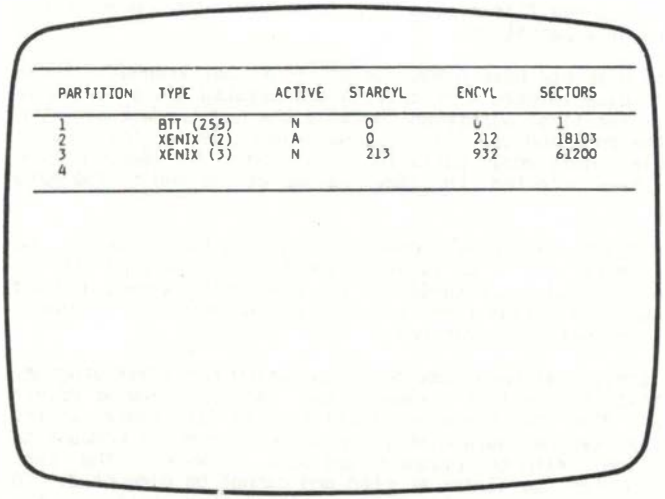
fdisk displays the following prompt:

```
Type ? for help
*
```

To create a new partition do the following:

1. Type the letter "p" and press CR . This selects the print command, which displays the current partition table for the selected

disk. For example:



PARTITION	TYPE	ACTIVE	STARCYL	ENDCYL	SECTORS
1	BTT (255)	N	0	0	1
2	XENIX (2)	A	0	212	18103
3	XENIX (3)	N	213	932	61200
4					

Check in the first column of the table that you have less than four partitions. Check also that you have cylinders available for a new partition, and where they reside.

2. Type the letter **c** and press **CR** . You are prompted as follows:

   Enter partition number [1-4]:

3. Type the number of the partition you wish to create and press **CR** . The number must not identify a partition already in use (if you wish to reemploy a partition already in existence you must use the "d" command to delete it first). Next you receive a prompt such as the following:

   Enter starting cylinder number [1-1022]:

   The available cylinder numbers depend on the capacity of the hard disk.

4. Enter the number of the starting cylinder for the partition and press **CR** . The starting cylinder must not be contained within another existing partition. You are further prompted:

   Enter length in 1K blocks:

5. Enter the length of the partition in 1K blocks and press **CR** For example, type "2000" to indicate a length of 2 Mbytes. The partition should not overlap any other existing partition.

The asterisk prompt will now return to the screen.

6. If you want to create a further partition on the same hard disk, select the "c" command again.

   When you have created all the partitions you want, select the "p" command again. The updated partition table will appear.

7. Check that each entry is correct. If you wish to modify an entry, you must select the "d" command to delete the relevant partition, then use the "c" command again as described above.

8. When all the entries are correct, type the letter "w" and press CR . This writes the new partition table to the hard disk.

9. Finally type the letter "q" and press CR . This exits fdisk and returns you to the console prompt.

ADDING A SECOND HARD DISK

If you add a second hard disk after installation time you must go through the following procedures before being able to use fdisk as described above:

1. Connect the hard disk unit (see the appropriate hardware instructions supplied with your M28 and the hard disk).

2. Run the SETUP option of the CUSTOMER TEST program to indicate to your XENIX system the presence of the additional hard disk. (See the M28 Installation and Operations Guide for full details of CUSTOMER TEST.)

3. Run the badtrack command as described in "Running badtrack" in Part 1 of this guide. badtrack must be invoked from the Installation diskette at boot time. It will prompt you to say whether you wish to overwrite the existing bad track table for your first hard disk. Type "n" and press CR to move straight on to the second hard disk.

You can now create partitions on your second hard disk.

CREATING A FILE SYSTEM

You can create a file system on a hard disk partition or formatted diskette by using the mkfs command. To create the file system on a disk you need:

- A formatted, blank diskette or an empty hard disk partition
- The special filename of the diskette or hard disk partition
- The block size of the diskette or hard disk partition, in 1K blocks

- The gap and block numbers for the disk

Special filenames for diskettes and hard disk partitions, possible block sizes and gap and block numbers are listed in Appendix A, "XENIX Special Device Files". (Note that the block size of a hard disk partition is always half the 512-byte sector size given in the last column of the partition table, as displayed by the "p" option of the fdisk program.)

Note that if a file system already exists on the target diskette or hard disk partition, it will be destroyed by this procedure. For this reason, be particularly careful not to create a new file system on the root file system. If you destroy the root file system, you will have to reinstall the XENIX system.

To make a file system, follow these steps:

1. Log in. You do not have to be logged in as the super-user to use the **mkfs** command.

2. If the target for the file system is a diskette, insert this now into a diskette drive. Make sure there is no read-only tab on the diskette jacket.

   If the target is a hard disk, make sure you know which hard disk partition is free for use.

3. Type:

   /etc/mkfs *specialfile blocksize gap block*

   (where *specialfile* , *blocksize* , *gap* and *block* are supplied by you) and press CR . Note that the *specialfile* must be a device name such as /dev/fd0, not a raw device name such as /dev/rfd0. The system automatically creates the file system. If it discovers data already on the diskette or hard disk partition, the system displays the message:

   mkfs: *specialfile* **contains** data. **Overwrite? (y/n):**

   If you are sure that the existing file system contains nothing you want to save, type y (for "yes") and press CR to overwrite the data and continue creating the file system. Otherwise, type n (for "no"). If you type n , no file system is created.

For example, the following command creates a file system on the diskette in drive 0 with blocksize 1200 and gap and block numbers 3 and 30:

```
/etc/mkfs /dev/fd0 1200 3 30
```

MOUNTING A FILE SYSTEM

Once you have created a file system, you can mount it with the mount command. To mount a file system you need:

- the special filename of the diskette drive or hard disk partition containing the file system you wish to mount
- the name of an empty directory to receive the file system

The special filenames of diskette drives and hard disk partitions are given in Appendix A, "XENIX Special Device Files". The directory to receive the file system may be any directory that is empty (contains no files) and is not your current working directory. Note that the directory /mnt is specifically reserved for mounted file systems.

To mount a file system, follow these steps:

1. Log in. You do not have to be logged in as the super-user to use this command.
2. If the file system is on diskette, insert it into a diskette drive.
3. Type the appropriate mount command and press CR . The command should have the form:

   /etc/mount *specialfile directoryname*

   where *specialfile* is the special filename of the diskette drive or hard disk partition containing the file system and *directoryname* is the name of the directory to receive the file system. If you are using a diskette with a read-only tab, you must include the switch -r at the end of the command.

For example, the following command mounts the diskette in drive 0 onto the directory named /account:

```
/etc/mount /dev/fd0 /account
```

Remember to make sure that the specified directory is empty before issuing the command. If the command displays the message:

```
mount: Structure needs cleaning
```

use the fsck command to clean the file system and try to mount it again (see "File System Integrity" in Chapter 6). If the command displays the message:

```
mount: mount failed
mount: Device busy
```

either the file system has already been mounted and cannot be mounted twice, or a user is currently in the directory. You must wait for users to leave a directory before you can mount the directory.

To check that the file system was properly mounted, use the cd command to change to the directory containing the mounted system and the ls -l command to list the contents. The command displays the files and directories in the file system. Be sure to use the cd command to leave the directory after finishing your work in it.

Note that frequently used file systems can be mounted automatically when starting the system by appending the appropriate mount commands to the /etc/rc file. See "Changing the /etc/rc File" in Chapter 10 for details.

UNMOUNTING A FILE SYSTEM

You can unmount a mounted file system with the umount command. Unmounting a file system does not destroy its contents. It merely removes access to the files and directories in the file system.

To unmount a mounted file system, use the syntax:

```
/etc/umount specialfile
```

Press CR . The *specialfile* is the name of the special file that corresponds to the diskette drive or hard disk partition that contains the file system. The command empties the directory that previously contained the file system and makes it available for mounting another file system.

For example, the following command unmounts a file system from diskette drive 0:

```
/etc/umount /dev/fd0
```

Before unmounting a file system, make sure that no files or directories are being accessed by other commands or programs. The umount command displays the message:

```
umount: unmount failed
umount: Device busy
```

if you or another user is currently in the directory containing the file system.

PERMISSIONS

Permissions control access to all the files and directories in a XENIX system. In XENIX, ordinary users may access those files and directories for which they have permission. All other files and directories are inaccessible.

There are three different levels of permissions: user, group, and other. User permissions apply to the owner of the file; group permissions apply to users who have the same group ID as the owner; and other permissions apply to all other users.

DISPLAYING PERMISSIONS

You can display the permission settings for all the files in a directory with the ls (for "list directory") command. This command lists the permissions along with the name of the file's owner, the size (in bytes), and the date and time the file was last changed. The command display has the following format:

```
-rw-rw---- 1 johnd  group  11515 Nov 17  14:21 file1
```

The permissions are shown as a sequence of ten characters at the beginning of the display. The sequence is divided into four fields. The first field (the "type" field) has a single character, the other fields ("user", "group" and "other") have three characters each. The characters in the fields have the following meanings.

In the type field:

d Indicates the item is a directory

- Indicates the item is an ordinary file

b Indicates the item is a device special block I/O file

c Indicates the item is a device special character I/O file

In the user, group and other fields:

r Indicates read permission. Read permission for a file means you may copy or display the file. Read permission for a directory means you may display the files in that directory.

w Indicates write permission. Write permission for a file means you may change or modify the file. Write permission for a directory means you may create files or subdirectories within that directory.

x Indicates execute permission (for ordinary files) or search permission (for directories). Execute permission for a file means you may invoke the file as you would a program. Execute permission for a directory means you may enter that directory with the cd command.

- Indicates no permission.

For example, the permissions:

```
-rwxrwxrwx
```

indicate an ordinary file with full read, write, and execute access for everyone (user, group, and other).

The permissions:

```
-rw-------
```

indicate an ordinary file with read and write access for the user only.

The permissions:

```
drwxr-x--x
```

indicate a directory with search access for everyone, read access for the user and group, and write access for only the user.

When you create a file, the XENIX system automatically assigns the following permissions:

```
-rw-r--r--
```

This means that everyone may read the file but only the user may write to it. When you create a directory, the system assigns the permissions:

```
drwxr-xr-x
```

This means everyone may search and read the directory but only the user may create and remove files and directories within it.

CHANGING PERMISSIONS

You can change the permissions of a file or a directory with the chmod (for "change mode") command. This command requires that you tell it how to change the permissions of a specific file or directory. You do so by indicating which levels of permissions you wish to change (user "u", group "g", or other "o"), how you wish to change them (add "+" or remove "-"), and which permissions you wish to change (read "r", write "w", or execute "x"). For example, the pattern:

```
u+x
```

adds execute permission for the user. The pattern:

```
go-w
```

removes write permission for group and other.

The chmod command has the form:

```
chmod pattern file ...
```

where *pattern* is the new level of permissions and *file* is the name of a file or directory. If more than one name is given, the names must be separated by spaces. For example, to change the permissions of the file "receivables" from "-rw-r--r--" to "-rw-------", type:

```
chmod go-r receivables
```

Press CR .

After using chmod , use the ls -l command to check the results. If you

have made a mistake, use chmod again to correct the mistake.

CHANGING THE FILE CREATION MASK

The file creation mask is a special number, kept by the system, that defines the permissions given to every file and directory created by a user. Initially, the mask has the value "022" which means every file receives the permissions:

```
-rw-r--r--
```

Every directory receives the permissions:

```
drwxr-xr-x
```

You can change the mask and the initial permissions your files and directories receive by using the umask command.

The umask command has the form:

```
umask value
```

where *value* is a three-digit number. The three digits represent user, group, and other permissions, respectively. The value of a digit defines which permission is given, as shown in the following list:

DIGIT	Permission
0	Read and write (also execute for directories)
1	Read and write
2	Read (also execute for directories)
3	Read
4	Write (also execute for directories)
5	Write
6	Execute for directories
7	No permissions

For example, the command:

```
umask 177
```

sets the file creation mask so that all files and directories initially have read and write permission for the user and no permissions for all others.

MANAGING FILE OWNERSHIP

Whenever a file is created by a user, the system automatically assigns user ownership of that file to that user. This allows the creator to access the file according to the user permissions. The system also assigns a group ownership to the file. The group ownership defines which group may access the file according to the group permissions. The group is the same group to which the user who created the file belongs.

Only one user and one group may have ownership of a file at any one time. (These are the owner and group displayed by the ls command.) However, you may change the ownership of a file by using the chown and chgrp commands.

CHANGING USER OWNERSHIP

You can change the user ownership of a file with the chown command. The command has the form:

chown *login-name file ...*

where *login-name* is the name of the new user and *file* is the name of the file or directory to be changed. For example, the command:

```
chown johnd projects.june
```

changes the current owner of the file projects.june to johnd.

The chown command is especially useful after you change the user ID of a user account (see "Changing a User ID" in Chapter 4).

You must be logged in as the super-user to use this command.

CHANGING GROUP OWNERSHIP

You can change the group ownership of a file with the chgrp command. The command has the form:

chgrp *group-name file ...*

where *group-name* is the name of a group given in the /etc/group file and *file* is the name of the file you wish to change. For example, the command:

```
chgrp shipping projects.june
```

changes the group ownership of the file projects.june to the group named shipping.

The chgrp command is especially useful if you have changed the login group of a user (see "Changing a User's Login Group" in Chapter 4).

SYSTEM SECURITY

Every system, no matter what its size, should have some form of protection from unauthorized access to the computer, disks and system files. The following sections suggest ways for a system administrator to protect the system.

PHYSICAL SECURITY

You can protect the physical components of the computer, especially system diskettes, by taking these steps:

1. Keep nonessential personnel out of the work area.
2. Organize and lock up all diskettes when not in use. They should not be stored with the computer itself.
3. Keep diskettes away from magnetic forces, direct sunlight, and severe changes in temperature.
4. Do not use ball-point pens to write labels on diskettes.
5. Make backup copies of all diskettes (see "Copying Diskettes" later in this chapter).

ACCESS SECURITY

You can protect the system from access by unauthorized individuals by taking these steps:

1. Remind users to log out of their accounts before leaving the terminal.
2. Discourage users from choosing passwords that are easy to guess. Passwords should be at least six characters long and include letters, digits, and punctuation marks.
3. Keep the super-user password secret from all but necessary personnel.

PROTECTING SPECIAL FILES

You can prevent ordinary users from gaining direct access to the data and program files on the system's hard disk(s) and diskettes by protecting the system's special files. The XENIX special files, in the /dev directory, are used primarily by the system to transfer data to and from the computer's hard disk(s) and diskettes and other devices, but can also be used by ordinary users to gain direct access to these devices.

Since direct access bypasses the system's normal protection mechanisms and allows ordinary users to examine and change all files in the system, it is wise to protect the special files to ensure system security.

To protect the XENIX special files, log in as the super-user and use the chmod command to set appropriate permissions. For example, to disallow any access by ordinary users, set the permissions of such special files as /dev/mem, /dev/kmem, /dev/root and /dev/usr to read and write access for the user only. Note that you must not change the permissions for the /dev/tty files.

COPYING DISKETTES

To ensure against the loss of data stored on diskettes, you can use the dd command to make copies of diskettes on new, formatted diskettes.

To make a copy of a diskette on a system with two diskette drives, follow these steps:

1. Insert the diskette to be copied into drive 0.

2. Insert an empty, formatted diskette into drive 1. If necessary, you can format a diskette with the format command described earlier in this chapter.

3. Type:

   dd if=/dev/fd0 of=/dev/fd1 count= *blkcount*

   Press CR . The *blkcount* must be the number of blocks on the diskette you are copying (see Appendix A, "XENIX Special Device Files" for details).

The command copies the first diskette to the second, then displays a record of the number of blocks copied.

To duplicate a diskette on a system with only one diskette drive, you must involve the hard disk. Follow these steps:

1. Insert the diskette to be copied into diskette drive 0.

2. Type:

   dd if=/dev/fd0 of= *filename* count= *blkcount*

   Press CR . The *filename* can be any empty file you wish to receive the transferred data. (If the file does not already exist it will be created.) The *blkcount* must be the number of blocks on the diskette you are copying (see Appendix A, "XENIX Special Device Files" for details).

3. Replace the source diskette with the formatted diskette you wish to copy to.

4. Type:

   dd if= *filename* of=/dev/fd0 count= *blkcount*

The *filename* and *blkcount* must be the same as in the previous command line.

5. Once you have made the transfer you can use the rm command to remove the data from the hard disk.

USING XENIX ACCOUNTING FEATURES

The XENIX system provides a set of commands that allow the system administrator to perform process accounting. Process accounting is a simple way to keep track of the amount of time each user spends on the system. The process accounting commands keep a record of the number of processes (i.e., programs) started by a user, how long each process lasts, and other information such as how often the process accesses I/O devices and how big the process is in bytes.

Process accounting is helpful on systems where users are being charged for their access time, and it may also be used to develop a detailed record of system, command, and system resource usage.

There are several commands that may be used to do process accounting. Of these, the most useful are accton and acctcom . The accton command starts and stops process accounting. When invoked, the command copies pertinent information about each process to the file named /usr/adm/pacct. You may use the acctcom command to display this information. The command has several options for displaying different types of accounting information.

STARTING PROCESS ACCOUNTING

You can start process accounting at any time, but it is typically started when the system itself is started. You start process accounting with the accton command. Type:

```
accton /usr/adm/pacct
```

The command automatically creates a new file /usr/adm/pacct and begins to copy process accounting information to it. If the /usr/adm/pacct file exists before starting accton , the file contents are deleted.

Note that when you start the system, the contents of the /usr/adm/pacct file is usually saved in the file /usr/adm/opacct.

DISPLAYING ACCOUNTING INFORMATION

The acctcom command reads processing accounting information from the /usr/adm/pacct file by default, then displays selected information on your terminal screen. The command usually displays basic accounting information, such as the process's program name, the name of the user who invoked the process, the start and stop times for the process, and the number of execution seconds in real time and CPU time. The command has several options you can select to display other information.

To display the average size of each process, type:

```
acctcom
```

The command displays the basic information plus the average size of each process.

To display basic accounting information about a specific command, use the syntax:

```
acctcom -n command
```

where *command* is the name of the command you are interested in. The command responds by displaying each entry for the specified command. For example, to display each entry for the system command units, type:

```
acctcom -n units
```

To display information about the number and size of input and output counts, type:

```
acctcom -i
```

The command displays basic program information plus the number of characters and blocks transferred or read by each program.

To display information about a program's use of system resources, type:

```
acctcom -h
```

The command displays the basic information plus the use factor. The use factor is a number generated and used by the system to determine how each process should be scheduled for execution. Processes with high use factors use a high percentage of the system resources and are therefore scheduled after processes with lower factors.

6. MAINTENANCE FILE SYSTEMS

ABOUT THIS CHAPTER

This chapter explains how to maintain file systems.

CONTENTS

INTRODUCTION

File system maintenance, an important task of the system administrator, keeps the XENIX system running smoothly, keeps the file systems clean, and ensures adequate space for all users. To maintain the file systems, the system administrator must monitor the free space in each file system, and take corrective action whenever it gets too low.

This chapter explains the file system maintenance commands. These commands report how much space is used, locate seldom-used files, and remove or repair damaged files.

MAINTAINING FREE SPACE

The XENIX system operates best when at least 15% of the space in each file system is free. In any system, the amount of free space depends on the size of the disk that contains the file system and the number of files on that disk. Since all disks have a fixed amount of space, it is important to carefully control the number of files stored on the disk.

If a file system has less than 15% free space, system operation usually becomes sluggish. If no free space is available, the system stops any attempts to write to the file system. This means that the user's normal work on the computer (creating new files and expanding existing ones) stops.

The only remedy for a file system that has less than 15% free space is to delete one or more files from the file system. The following sections describe strategies for keeping the free space available.

STRATEGIES FOR MAINTAINING FREE SPACE

The system administrator should regularly check the amount of free space of all mounted file systems and remind users to keep their directories free of unused files. You can remind users by including a reminder in the message of the day file /etc/motd. (See the section in Chapter 10, entitled "Changing the /etc/motd File".)

If the amount of free space slips below 15%, the system administrator should:

1. Send a system-wide message asking users to remove unused files.

2. Locate exceptionally large directories and files and send mail to the owners asking them to remove unnecessary files.

3. Locate and remove temporary files and files named core.

4. Clear the contents of system log files.

Finally, if the system is chronically short of free space, it may be necessary to create and mount an additional file system.

DISPLAYING FREE SPACE

You can find out how much free space exists in a particular file system with the df (for "disk free") command. This command displays the number of blocks available on the specific file system. A block is 512 characters (or bytes) of data.

The df command has the form:

```
df specialfile
```

where *specialfile* can be the name of a XENIX special file that corresponds to the disk drive that contains the file system (see Appendix A, "XENIX Special Device Files"). If you do not give a special filename, then the free space of all normally mounted file systems is given.

For example, to display the free space of the root file system /dev/root, type:

```
df /dev/root
```

and press . The command displays the special filename and the number of free blocks.

You can most easily arrive at the percentage of free space by using the df -v option. This reports the percentage of blocks already used.

SENDING A SYSTEM-WIDE MESSAGE

If free space is low, you may send a message to all users on the system with the wall (for "write to all") command. This command copies the messages you type at your terminal to the terminals of all users currently logged in.

To send a message, type:

```
wall
```

and press CR . Type the message, pressing CR to start a new line if necessary. After you have typed the message, press CR and then CTRL D . The command displays the message on all terminals in the system and returns to the system prompt.

DISPLAYING DISK USAGE

You can display the number of blocks used within a directory by using the du command. This command is useful for finding excessively large directories and files.

The du command has the form:

```
du directory
```

The optional *directory* must be the name of a directory in a mounted file system. If you do not give a directory name, the command displays the number of blocks in the current directory.

For example, to display the number of blocks used in the directory /usr/johnd, type:

```
du /usr/johnd
```

and press CR . The command displays the name of each file and directory in the /usr/johnd directory and the number of blocks used.

DISPLAYING BLOCKS BY OWNER

You can display a list of users and the number of blocks they own by using the quot (for "quota") command. The command has the form:

```
quot specialfile
```

The *specialfile* must be the name of the special file that corresponds to the disk drive that contains the file system (see Appendix A, "XENIX Special Devices Files").

For example, to display the owners of files in the file system mounted on the disk drive /dev/fd1 , type:

```
quot /dev/fd1
```

and press CR . The command displays a list of the users who have files in the file system and the number of blocks in these files.

MAILING A MESSAGE TO A USER

If a particular user has excessively large directories or files, you may send a personal message to the user with the **mail** command.

To begin sending a message through the mail, use the form:

```
mail login-name
```

and press CR . The *login-name* must be the login name of the recipient. To send a message, type the message, press CR and then press CTRL D . If the message has more than one line, press CR at the end of each line. The **mail** command copies the message to the user's mailbox, where the user may view it by using the mail command. See the XENIX User Guide for details.

LOCATING FILES

Using the find command, you may locate files that have a specified name, size, date, owner, and/or last access date. The find command is useful for locating seldom-used and excessively large files.

The find command has the form:

find *directory parameters*

The *directory* must be the name of the first directory you want to search. (The command will also search all subdirectories within that directory.) The *parameters* are special names and values that tell the command what to search for (see find(C) in the XENIX User and System Administrator Reference Manual for complete details). The most useful parameters are:

-name *file*

-atime *number*

-print

The -name parameter causes the command to look for the specified file. The -atime parameter causes the command to search for files which have not been accessed for the number of days. The -print parameter causes the command to display the locations of any files it finds.

For example, to locate all files named core in the directory /usr, type:

find /usr -name core -print

and press CR . The command displays the locations of all files it finds.

LOCATING CORE AND /TMP FILES

You can locate core and /tmp (temporary) files with the find command.

A core file contains a copy of a terminated program. The XENIX system sometimes creates such a file when a program causes an error from which it cannot recover. A /tmp (temporary) file contains data created as an intermediate step during execution of a program. These files may be left behind if a program contained an error or was prematurely stopped by the user. The name of a /tmp file depends on the program that created it.

In most cases, the user has no use for either core or /tmp files and they can be safely removed.

When you search for core or /tmp files, it is a good idea to search for files which have not been accessed for a reasonable period of time. For example, to find all core files in the /usr directory which have not been accessed for a week, type:

find /usr -name core -atime+7 -print

and press CR .

CLEARING LOG FILES

The XENIX system maintains a number of files, called log files , that contain information about system usage. When new information is generated, the system automatically appends this information to the end of the corresponding file, preserving the file's previous contents. This means the size of each file grows as new information is appended. Since the log files can rapidly become quite large, it is important to periodically clear the files by deleting their contents.

You can clear a log file by typing:

cat */dev/null filename*

where *filename* is the full pathname of the log file you wish to clear. A log file normally receives information to be used by one and only one program, so its name usually refers to that program. Similarly, the format of a file depends on the program that uses it.

In some cases, clearing a file affects the subsequent output of the corresponding program. For example, clearing the file /etc/ddate forces the next backup to be a periodic backup (see Chapter 7, "Backing Up File Systems"). A list of log files appears in Appendix B, "XENIX Directories".

EXPANDING THE FILE SYSTEM

If free space is chronically low, it may be to your advantage to expand the system's storage capacity by creating and mounting a new file system. When you have mounted the file system, you may use this new file system for your work, or to copy user or system directories to it.

A chronic shortage of space usually results from having more users on the system than the current hard disk can reasonably handle, or from having too many directories or files. In either case, creating a new file system allows some of the users and directories to be transferred from the hard disk, freeing a significant amount of space on the existing file system and improving system operation. For details about creating and mounting file systems, see Chapter 5, "Using File Systems".

FILE SYSTEM INTEGRITY

Since file systems are normally stored on hard disks and diskettes occasional loss of data from the file system through accidental damage to the disks is not unusual. Such damage can be caused by conditions such as an improper system shutdown, hardware errors in the disk drives, or a worn out disk.

Such damage usually affects one or two files, making them inaccessible. In very rare cases, the damage causes the entire file system to become

inaccessible.

The XENIX system provides a way to restore and repair a file system if it has been damaged. The fsck (for "file system check") command checks the consistency of file systems and, if necessary, repairs them. The command does its best to restore the information required to access the files, but it cannot restore the contents of a file once they are lost. The only way to restore lost data is to use backup files. For details about backup disks, see Chapter 7, "Backing Up File Systems".

REPAIRING THE FILE SYSTEM

You can repair a file system with the fsck command. The command has the form:

```
fsck specialfile
```

The *specialfile* must be the name of the special file that corresponds to the disk drive that contains the file system (see Appendix A, "XENIX Special Device Files").

For example, to check the file system on the disk in the disk drive /dev/fd1, type:

```
fsck /dev/fd1
```

and press CR . The program checks the file system and reports on its progress with the following messages:

```
** Phase 1 - Check Blocks and Sizes
** Phase 2 - Pathnames
** Phase 3 - Connectivity
** Phase 4 - Reference Counts
** Phase 5 - Check Free List
```

If a damaged file is found during any one of these phases, the command

asks if it should be repaired or salvaged. Type **y** (for "yes") to repair a damaged file. You should always allow the system to repair damaged files, even if you have copies of the files elsewhere or intend to delete the damaged files.

Note that the fsck command deletes any file that it considers too damaged to be repaired. If you suspect a file system problem and wish to try to save some of the damaged file or files, check other possible remedies before you invoke the command.

AUTOMATIC FILE SYSTEM CHECK

The XENIX system sometimes requests a check of the file system when you first start it. This usually occurs after an improper shutdown (for example, after a power loss). The file system check repairs any files disrupted during the shutdown. For details, see "Cleaning the File System" in Chapter 3.

7. BACKING UP FILES

ABOUT THIS CHAPTER

This chapter explains how to back up files.

CONTENTS

INTRODUCTION

A backup is a diskette or cartridge tape copy of files that are in the root and usr directories and in other regularly mounted file systems. A backup allows the system administrator to save a copy of a file system or individual file as it is at a specific time. The copy may be used later to restore a file or files that are accidentally lost or temporarily removed from the file system to save space.

This chapter explains how to create backups and how to restore files from the backups.

STRATEGIES FOR BACKUPS

The system administrator should back up files on a regular basis. In particular, you should make daily copies of all files that have been modified during the day and you should make periodic (for example, weekly) copies of the entire root and usr file systems and any other regularly mounted file systems.

The XENIX system offers several ways to back up files. The most versatile backup commands are tar and cpio . Both these commands can be used to back up individual files and directories. They can be used with both diskettes and cartridge tapes. They can be used by ordinary users as well as the system administrator and are recommended for small-scale backups.

If you want to back up complete file systems you can use in addition to these commands either backup (with the correlative restore), or sysadmin. sysadmin is a menu-driven utility especially written for file system backups on diskettes. It does not work with cartridge tapes. You must be logged in as super-user to use this program. backup works with both diskettes and cartridge tapes and offers the fastest way to back up substantial file systems. You do not have to be logged in as super-user to use backup or restore .

Each of these commands is summarized in this chapter. See the XENIX User and System Administrator Reference Guide for more information on them.

Note that you can also use the mkfs command to make a new file system on a disk. Such a file system can then be mounted and its pathname given as a backup destination in place of a device name. See Chapter 5 for more on the mkfs command.

A typical backup schedule includes a daily backup and a periodic backup once a week. A daily backup copies only those files modified during that day; a periodic backup copies all files in the file system. The appropriate schedule for a system depends on how heavily the system is used and how often files are modified. In all cases, a periodic backup should be done at least once a month.

The system administrator should schedule backups at times when few (if any) users are on the system. This ensures that the most recent version of each file is copied correctly.

A regular schedule of backups requires a good supply of diskettes or tape cartridges. Daily backups should be saved for at least two weeks; periodic backups should be saved indefinitely. Diskettes and tape cartridges should be properly labeled with the date of the backup and the names of the files and directories contained in the backup. After a backup has expired, it may be overwritten by a new backup.

Diskettes intended for backup must be formatted in advance. Chapter 5 of this guide gives details of how to do this. Tape cartridges must also be formatted prior to use. The following section describes the tape formatting procedure. Note also that if you add a cartridge tape drive to your system after your initial installation, you must reconfigure your system to include the new addition. See Chapter 9 for the procedures involved.

FORMATTING TAPE CARTRIDGES

To format a tape cartridge you must use the tpformat program. Follow these steps:

1. Insert a cartridge tape in your tape drive.

2. Type:

   ```
   tpformat
   ```

   and press CR . The program is invoked and the message:

   ```
   Ready to Format, Continue ? (y/n) :
   ```

   appears on the screen.

3. Press CR . Assuming your tape has not previously been formatted, the following message appears:

   ```
   TAPE NOT SERVO WRITTEN, Servowrite ? (y/n) :
   ```

4. Type **y** and press CR to servo write the tape. Servo writing is a low-level format operation that need only be performed on virgin tapes or on tapes on which a bulk erasor has been used. (If the tape does not fall into one of these categories the prompt will not appear.)

   A message such as the following appears:

   ```
   Servo writing (approximately 20 minutes)...
   ```

5. The tape gets servo written. This takes approximately 20 minutes for a 125 model tape drive, 18 minutes for a 110 model.

   When the servo write is complete, the following messages appear:

   ```
   Servo write completed
   ```

```
Ready to Format continue ? (y/n) :
```

6. Press CR to continue with the tape format operation. Information about the areas of the tape being formatted and then verified appears on the screen. A bad block list is compiled and a tape parameters block is created and verified. The whole operation takes about 20 minutes for a 125 model drive, 16 for a 110 model. When it has finished the XENIX prompt returns to the screen.

USING THE TAPE INFORMATION (TPINFO) PROGRAM

You can use the tpinfo program to display and in certain cases modify information contained in the tape parameters block of a cartridge tape.

Whenever you do a backup to tape, a separate volume or "save set" is created. You can thus build up a library of different backups on a single tape. When you come to restore files from a particular backup, tpinfo lets you identify the save set you require.

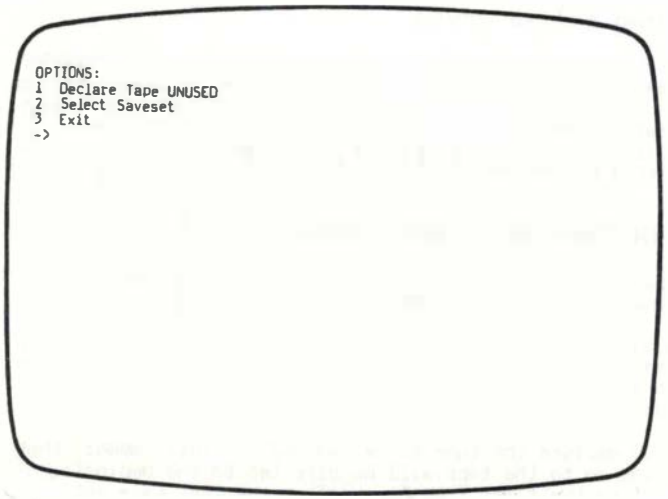
tpinfo also lets you declare the tape to be "unused". This means that the next backup you do to the tape will be directed to the beginning of the tape, instead of to the first free block after the last save set.

To invoke tpinfo , type:

```
tpinfo
```

and press CR . Information about the size of the tape and any save sets that it contains appears, along with the number of Kbytes of storage still available.

The following menu also appears:



Type:

1

and press CR to set the tape to unused.

Type:

2

and press CR to select a save set. The following message appears:

Saveset Number ? >

Type the number of the save set you wish to select and press CR . Once you have selected the save set of your choice you can use one of the backup/restore commands to restore part or all of the backed-up volume.

Type

3

and press CR to return to the XENIX prompt.

Note that neither tpinfo nor any of the backup commands gives you the opportunity to identify your save sets by name. You should therefore use a label on the tape itself, or maintain a file on your system, to remind yourself of the contents of the various save sets.

USING THE SYSADMIN PROGRAM

The sysadmin program is a formal maintenance program for systems that require a rigorous schedule of file system backups. Such systems usually have many users and a large number of files that are modified daily. The program automatically locates modified files, copies them to diskette, and optionally produces a list of the files.

The sysadmin program performs daily and periodic backups of both root and user file systems, lists backup files and restores individual files from backup diskettes. The program presents each task as an item in a menu. To perform a task, simply choose the appropriate item from the menu and supply the required information.

CREATING BACKUPS

To create backups with the sysadmin program, you need several formatted diskettes. The exact number depends on the number of files you want to copy; for example, some periodic backups require as many as nine diskettes. For details on how to format a diskette, see the section in Chapter 5 entitled "Formatting Diskettes".

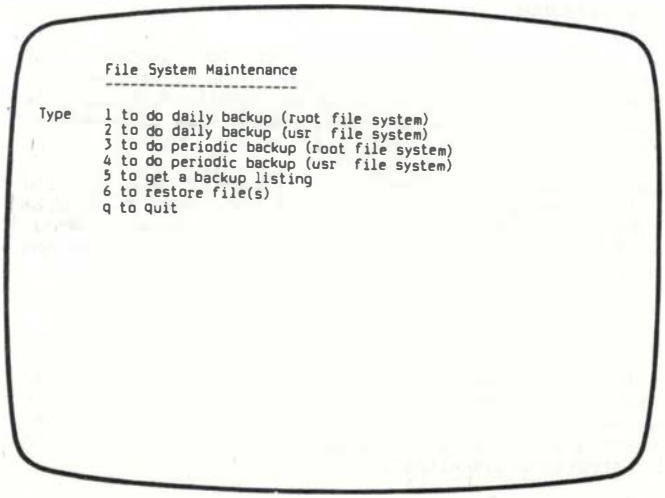
To create a backup, follow these steps:

1. Log in as the super-user.

2. Type:

   sysadmin

   and press CR . The program displays a file system maintenance menu:



3. Type 1 or 2 for a daily backup or 3 or 4 for a periodic backup. Then press CR . Note that if the system has never had a periodic backup, it automatically performs one, even if you have chosen a daily backup.

   The following message appears:

   Insert first disk in drive zero, then press ENTER:

4. Insert a diskette in drive 0, wait for the drive to accept the diskette (all drive noise should stop), and press CR . The program displays the current date and the date of the last backup (it displays "the epoch" if there has been no backup). It scans the file system to be backed up and displays its estimate of the number of volumes (that is, diskettes) required to complete the operation. It then begins to copy files to the diskettes. If the diskette runs out of space, the program displays the message:

   Please insert new volume, then press ENTER:

5. Remove the first diskette and insert a new diskette. Wait for the drive to accept the diskette, then press CR . The program continues to copy files to the new diskette. Repeat this step until the program displays the message:

   DONE

You should label each diskette as you remove it from the drive. For example, label the first diskette "Volume 1," the second "Volume 2," and so on.

GETTING A BACKUP LISTING

You can keep a record of the files you have backed up by invoking the sysadmin program and selecting choice 5 on the menu. The program copies the names of all files from the backup diskettes to the temporary file /tmp/backup.list. This listing is especially convenient if you keep detailed records of the files copied in each backup. The backup listing is available after every daily or periodic backup.

To get the listing, follow these steps:

1. Log in as the super-user.

2. Type:

   sysadmin

   and press CR . The program displays the system maintenance menu.

3. Type 5 and press CR . The program asks you to reinsert the backup diskettes in the same order that you inserted them during the backup.

4. Insert the first diskette, wait until the drive accepts the diskette, then press CR . The program automatically reads the filenames off the backup diskette and places them in the list file. When the program has read all the names, it asks for the next diskette.

5. Remove the first diskette and insert the next. Wait for the drive to accept the diskette and press CR . Repeat this step until all diskettes have been read.

You may produce a printed copy of the backup list by printing the list at the lineprinter. Type:

```
lpr /tmp/backup.list
```

and press CR . To save space after printing the file, you should remove it from the /tmp directory with the rm command.

RESTORING A BACKUP FILE

You can restore files from the backup disks by invoking the sysadmin program and selecting choice 6 on the menu. You will need the complete set of backup diskettes that contains the latest version of the file you wish to restore. You will also need the full pathname of the file you wish to restore. This is the name given for the file in the backup listing.

To restore a file, follow these steps:

1. Log in as the super-user.

2. Type:

   sysadmin

   and press CR . The program displays the file system maintenance menu.

3. Type 6 and press CR . The program asks you to type the full pathname of the file you wish to restore.

4. Type the pathname and press CR . The program asks for another pathname.

5. Repeat step 4 to enter another pathname, or press CR to continue the program. If you press CR , the program asks you to insert the first diskette in the backup set.

6. Insert the first diskette in the set of backup diskettes (volume 1), wait for the drive to accept the diskette, and press CR . The program displays the inode numbers of the files you have given, then asks for the volume number of the backup diskette that contains the files.

7. Insert the diskette that has the correct volume number, type the volume number, and press CR . The program searches the diskette for the specified files. If found, the files are copied to your current directory. If not found, the program asks for the next volume.

8. Repeat step 7 until all files have been found and copied.

USING THE BACKUP AND RESTORE COMMANDS

The backup command offers a powerful and flexible tool for doing file-system backups on both diskettes and cartridge tapes. It allows you to maintain a set of backups of different "backup levels" and so ensure both complete backup cover and up-to-the minute archiving. Files backed up using backup can be restored by means of the restore command.

USING BACKUP

The syntax of the backup command is as follows:

backup [*key* [*arguments*] *filesystem*]

By default, backup takes as input and output the file system and device listed in the /etc/default/backup file. If you are not sure what these defaults are, use:

cat /etc/default/backup

to display the file's contents. If you wish to use input or output other than the default, you can either use an editor to modify the default file

or specify a file system or device name on the command line. For example, the following command line directs backup to back up a usr file system to a cartridge tape drive:

```
backup f /dev/rtpnv /dev/usr
```

In the above example, no backup level is indicated in the key. In this case the default values 9u are applied to the command line. The number 9 indicates the level of backup to be performed and represents the least comprehensive, most short-term type of back up. The "u" indicates that the /etc/ddate file will be updated with the date and level of the current back up.

To institute a regular back-up routine, you should start at the most comprehensive level of backup. This is level 0, which backs up all files in the given file system regardless of any entry in the /etc/ddate file.

You should then proceed with backups of differing levels. For example, you could do a level 2 backup every other week, interspersed with levels 0 and 1. The schedule for a single month might look something like this:

Date	level 2	level 1	level 2	level 0	level 2	level 1
Jan 1	root	usr				
Jan 8		root	usr			
Jan 15			root	usr		
Jan 22				root	usr	
Jan 29					root	usr

Provided you specify "u" each time to update the /etc/ddate file, a series of backups of this kind produces archives of all the files modified during a single week (the "2u" backups) together with periodic backups ("1u") and the occasional total backup ("0u").

High-level backups (such as "9u") are usually done on a daily basis. They will pick up and archive only the files that have been created or modified since a previous backup at a lower level.

You can do backups at times when no users are likely to be on the system by including backup within a crontab file. See the XENIX User and System Administrator Reference Guide for more information.

USING RESTORE

Use restore to restore files and file systems backed up with the backup command. The syntax of the command is as follows:

```
restore key [ arguments ]
```

Use the key character "f" followed by a device name to indicate a backup device other than the default. Use the key character "x" followed by one or more filenames to restore individual files. Use the key character "r" to restore a complete archive. In this case you should be sure to create a clear file system to receive the archive.

Suppose you wanted to restore the contents of a cartridge tape. Assuming you had a fourth partition of 5000 Kbyte blocks set up on your hard disk, and the contents of the tape were less than that, you could type:

```
/etc/mkfs /dev/hd01 5000 1 34
restore fr /dev/rtpnv /dev/hd01
```

(Note that the device file name /dev/hd01 represents the fourth, not the first, partition on the hard disk. See Appendix A for a complete list of special device file names.)

USING THE TAR COMMAND

The tar command copies specified files and directories to and from diskettes and tapes. On systems with one or two users, it gives the system administrator a direct way to make backup copies of the files that were modified during a day. On systems with many users, it gives ordinary users a way to make personal copies of their own files and directories.

COPYING FILES TO A TAR DISKETTE OR TAPE

You can copy individual files or directories to a diskette or tape with the tar command. The command has the form:

```
tar cvfk specialfile files
```

The cvfk options select the functions tar will perform. The *specialfile* must be the name of the special file that corresponds to a diskette or tape drive (see Appendix A, "XENIX Special Device Files"). The drive must contain a formatted diskette or tape, as the case may be. The *files* are the names of the files or directories you wish to copy.

For details about how to format a diskette, see "Formatting Diskettes" in Chapter 5. Information on formatting cartridge tapes is given earlier in this chapter. If you give a directory name, the command copies all files in the directory (including subdirectories) to the diskette or tape. For more information on tar options, see tar(C) in the XENIX User and System Administrator Reference Manual.

For example, to copy the files a, b, and c to the diskette in drive /dev/fdl, type:

```
tar cvfk /dev/fdl 360 a b c
```

and press **CR** . The value "360" is related to the "k" switch, and indicates the capacity in Kbytes of the diskette in the specified drive. This switch setting tells tar to pause each time a diskette is filled, to allow you the opportunity to replace it with a fresh one.

RESTORING FILES FROM A TAR DISKETTE OR TAPE

You may also use the tar command to restore files from a diskette or tape. The command copies all files on the diskette or tape "save set" to your current directory. In this case, the command has the form:

```
tar xvf specialfile
```

The xvf options select the functions tar will perform. The *specialfile* must be the name of the special file that corresponds to the drive that contains the tar diskette or tape.

For example, to restore files from the diskette in drive /dev/fdl, type:

```
tar xvf /dev/fdl
```

and press **CR** . The command copies files on the diskette in the drive to the current directory.

Since the tar command copies files only to the current directory, make sure you are in the desired directory before you invoke the command. You can change to the desired directory with the cd command.

USING THE CPIO COMMAND

You can also use the cpio command to back up individual files and directories. cpio reads the standard input and output, and thus is commonly used in conjunction with pipes and input/output redirection signs to produce meaningful results.

To copy out to an archive, use the following syntax:

```
cpio -o [ key ]
```

To copy back in from an archive, use the following syntax:

```
cpio -i [ key ] [ patterns ]
```

To copy a directory structure to another destination on the same input/output device, use the following syntax:

```
cpio -p [ key ] directory
```

Here is an example of a procedure that backs up the (sorted) contents of directory /usr/bin to tape:

```
cd /usr/bin
find . -print|sort|cpio -ocBv > /dev/rtpnv
```

The command line:

```
cpio -icBtv < /dev/rtpnv
```

displays the contents of the archive thus created, and is the equivalent to doing an `ls -l .`

8. SETTING UP THE CONSOLE, TERMINALS AND PRINTERS

ABOUT THIS CHAPTER

This chapter explains how to set up your console, terminals and printers.

CONTENTS

INTRODUCTION

One important task of the system administrator is to make sure that all the hardware elements of your M28 system are set up correctly. This may involve not only the original elements of your system, but any peripheral devices such as terminals, hard disks, cartridge tape drives or printers that may be added at a later date.

This chapter summarizes the steps you need to go through whenever you add a device to your system. It then explains how to use XENIX commands to set up and maintain terminals and printers. It also tells you how to configure your system console keyboard for foreign language and programmed function key support. For information on how to set up a second hard disk, see Chapter 5, "Using File Systems". For information on how to set up a cartridge tape drive, see Chapter 7, "Backing up Files".

Note that all physical connections between a device and the system are device-dependent. For information about these connections, see the hardware manuals provided with the device and your M28.

ADDING A DEVICE

Adding a peripheral device lets more users use the system, gives extra storage space for users' files and directories, and adds to the system's overall capabilities.

To add a peripheral device:

1. Make the physical connection between the device and the computer. Refer to the hardware instructions provided with the device.

2. Run the SETUP option of the CUSTOMER TEST utility in order to notify the system of the addition (see the M28 Installation and Operations Guide for full details of CUSTOMER TEST).

3. Use the correct system commands to set up the device for operation.

4. If the device is of a type not previously configured into your system by means of the appropriate configuration file, you must reconfigure your XENIX kernel accordingly. See Chapter 9 for more details of this procedure.

ADDING A TERMINAL

Extra terminals allow two or more users to use the system simultaneously. Many different terminals work well with the XENIX operating system. A short list of recommended models is given in terminals(M) in the XENIX User and System Administrator Reference Manual.

In addition to standard terminals, you can connect Personal Computers to your XENIX system providing they are set to terminal emulation mode. To set an Olivetti PC to terminal emulation mode, you should use the

(separate) Olitalk package. See the Olitalk User Guide for full details.

Before you can add a terminal, you must know how to connect the terminal to a serial line on the computer. You also need to know the name of the serial line. Physical connections for the terminal should be explained in the terminal's hardware manual. You should also consult the documentation available with the modems you are using. If you are connecting directly to the serial line you will need a null modem. The names of the available serial lines on the system are given in Appendix A, "XENIX Special Device Files".

Once a terminal has been connected, you may usually go ahead and enable the terminal for use with the enable command. However, there are additional initial settings to make if the terminal you are adding is non-standard, or if you are adding more than two terminals to your system. These are described in the following two subsections.

MODIFYING THE /ETC/TTYS FILE

The /etc/ttys file is read by the init program whenever you boot up the system and select normal operation. It contains information about each of the terminal devices connected to the system.

The default /etc/ttys entries are as follows:

```
1Cconsole
06tty00
06tty01
```

The first digit indicates whether or not the terminal is enabled for logins. Whenever init encounters a terminal entry with a "1" prefix, it creates a process that opens the serial line. The "0" prefix on the other hand indicates that the terminal is disabled -- in this case it is as though the terminal did not exist.

Using an editor you can change this first character to enable or disable a terminal. Be careful not to disable the console, however. Note also that you can use the enable and disable commands (described later in this chapter) to fulfil the same purpose. These commands themselves modify the /etc/ttys file.

The second character in an /etc/ttys entry indicates the default line mode for the serial line (the letter "C" indicates that the terminal is the console, with no serial interface). When init has opened a line, it executes the getty program and passes the line mode entry to it. The job of getty is to identify the correct line mode. It does this by referring to a file called /etc/gettydefs. This file contains a series of entries for different terminal line characteristics, including baud rate. It first tries the entry identified by the number passed to it from /etc/ttys. If this entry does not produce a successful connection, it cycles through the remaining entries until it finds one that does. When this happens, the login prompt will appear.

For more on the /etc/gettydefs file, see later in this chapter. For now,

you should make sure that each terminal connected to your M28 has an entry in /etc/ttys and that each entry refers to a valid line mode setting (such as the default "6").

The /etc/ttys file can only be edited from within system maintenance mode.

MODIFYING THE /ETC/TTYTYPE FILE

The /etc/ttytype file contains an entry for each terminal (including the console) that you have connected (or may wish to add) to your system.

The first entry should always be "ansi console". This indicates that the M28 XENIX console is to operate in ansi mode.

The second and third entries, by default, are as follows:

```
vt100 tty00
vt100 tty01
```

This indicates that any connected terminals on serial lines tty00 and tty01 will be regarded as VT100s. These are the correct settings for Olivetti WS584 terminals. They may not, however, be correct for other types of terminal. Also, you may have more than two terminals connected.

To identify the correct terminal type for each terminal, you should look in the /etc/termcap file. You can search for the entry for your particular terminal with the following command:

```
grep vendor /etc/termcap
```

where *vendor* represents the name of the terminal's manufacturer. Look at the applicable entry (there may be more than one for the manufacturer, in which case identify the model you have). The terminal's type is listed at the left of the entry.

You can now go back to the /etc/ttytype file and, using an editor, modify the entry for the appropriate serial line (or add a new entry).

If you are unable to identify the terminal, you should use the name "dumb".

Note that the /etc/ttytype entry is an initial setting only. It may be that you are using more than one terminal on a single line (for example, if you are going through a modem). In this case users will need to identify at login-time the terminal they are using (see "Setting the Terminal Type" later in this chapter).

ENABLING A TERMINAL

To enable a terminal, follow these steps:

1. Using the recommended procedure in the terminal's hardware manual, connect the terminal to one of the serial lines on your M28.

2. Reboot the system (this is important if you have made any modifications to the /etc/ttys or /etc/ttytype file as described in the previous two subsections).

3. Log in as the super-user.

4. Use the enable command to enable the terminal. The command has the form:

   enable *specialfile*

   where *specialfile* is the name of the serial line to which the terminal is attached.

   For example, the command:

   enable /dev/tty01

   enables the terminal connected to serial line /dev/tty01.

5. Turn on the power to the terminal and press **CR** several times. The system should display a "login:" message. When it does, you may log in and begin work.

If no "login:" message appears on the screen, if random characters appear, or if the terminal does not respond to your attempt to log in, you may need to change the baud rate (line speed) of the serial line, using the stty command. You may also need to set the terminal type correctly, using the TERM assignment statement. These methods are described in the next two sections of this chapter.

When using the enable command, make sure that you wait a full minute between each use of the command. Failure to do so can cause a system crash.

CHANGING SERIAL LINE OPERATION

Whenever you enable a terminal with the enable command, the system automatically sets the operating characteristics of the serial line to a set of default values (specified in the /etc/gettydefs file). Sometimes these values do not match the values the terminal uses, so you must change them to allow communication between the system and the terminal. You can display and change the operating characteristics of a serial line with the stty (for "set tty") command.

You can display the current operating characteristics of a serial line by typing:

```
stty
```

at the terminal connected to that line. If it is impossible to log in at that terminal, you may use another terminal to display the characteristics. Log in as the super-user at another terminal, and type:

```
stty< specialfile
```

where *specialfile* is the name of the device special file that corresponds to the serial line (see Appendix A, "XENIX Special Device Files"). For example, the command:

```
stty </dev/tty01
```

displays the current characteristics of the serial line named /dev/tty01. The command displays the baud rate, the parity scheme, and other information about the serial line. The meaning of this information is explained in *stty(C)* in the XENIX User and System Administrator Reference Manual.

One common change to a serial line is to change the baud rate. This is usually done from a terminal connected to another serial line, since changing the rate disrupts communication between the terminal and system. Before you can change the rate, you need to know the current baud rate of the terminal (review the terminal's hardware manual to see how to determine the terminal's current rate). Once you have the baud rate, log in as the super-user at the other terminal, and type:

```
stty baud-rate < specialfile
```

where *baud-rate* is the terminal's current baud rate, and *specialfile* is the name of the device special file that corresponds to the serial line you wish to change. The baud rate must be in the set 50, 75, 110, 134, 150, 200, 300, 600, 1200, 2400, 4800, and 9600. For example, the command:

```
stty 9600 </dev/tty01
```

changes the baud rate of the serial line /dev/tty01 to 9600. Note that the "less than" symbol (<) is used for both displaying and setting the serial line from another terminal.

Another common change is to change the way the system processes input and output through the serial line. Such changes are usually made from the terminal that is connected to the serial line. For example, the command:

```
stty -tabs
```

causes the system to expand tabs with spaces (used with terminals which do not expand tabs on their own), and the command:

```
stty echoe
```

causes the system to remove a character from the terminal screen when you back over it with the BKSP key.

Note that the stty command may also be used to adapt a serial line to an unusual terminal, or to another type of serial device that requires parity generation and detection or special input and output processing.

For a full description of this command, see stty(C) in the XENIX User and System Administrator Reference Manual.

SETTING THE TERMINAL TYPE

The XENIX system requires that an enabled terminal's type be clearly defined before any work is done at the terminal. You can set the terminal type by assigning the type to the TERM variable, a special XENIX system variable that associates the terminal you are using with a list of characteristics given in the /etc/termcap file. The characteristics tell the system how to interpret your terminal's keys and how to display data on your terminal's screen.

The TERM assignment has the form:

```
TERM= termtype ; export TERM
```

The *termtype* must be one of the names associated with one of the terminals defined in the /etc/termcap file. The assignment must be typed at the terminal whose type you are setting.

For example, to set the terminal type to "ansi", go to the terminal you wish to set, type:

```
TERM=ansi ; export TERM
```

and press CR .

If you are not sure which name you may use for *termtype* , you can view the names by displaying the /etc/termcap file. You can search for the entry for your particular terminal with the following command:

```
grep vendor /etc/termcap
```

where *vendor* represents the name of the terminal's manufacturer. Look at the applicable entry (there may be more than one for the manufacturer, in which case identify the model you have). The terminal's type is listed at the left of the entry.

You can let the system define the terminal .profile type automatically whenever a user logs in by including the TERM assignment in his/her .profile file (see Chapter 8, "Solving System Problems").

If you do let the system set the terminal type, be careful if it is possible for the user to log in on terminals that are not the same as his normal terminal. The XENIX system has no way of checking whether or not the terminal assignment is correct for the given terminal, and assumes that it is the same as the normal terminal.

If you have a mixed assortment of terminal types on your XENIX system,

and you do not know which terminal a given user will be logging in at, you can include a procedure such as the following in his/her .profile file:

```
case `tty` in
*console) TERM=ansi;;
*tty00) TERM=vt100;;
*tty01) TERM=I9;;
esac
export TERM
```

In this example the terminal type will be set to "ansi" if the user logs in on the console, "vt100" if he logs in on the terminal connected to tty00, "I9" (the type associated with the IBM 3101 Terminal Emulator) if he logs in on tty01.

SETTING THE TERMINAL LINE DEFAULTS

Your XENIX system can adapt itself to several different terminal baud rates and settings. The same program that prints the login message, getty , reads these terminal line values from a table and tries each setting until one is successful. The user can then log in to the system. This table provides several default settings for different kinds of terminals lines. The login process invokes getty automatically.

The table of terminal settings is found in a file called /etc/gettydefs. You can edit gettydefs to add different sets of terminal characteristics or change the existing ones.

THE GETTYDEFS FILE

The file /etc/gettydefs contains the information that getty uses to set up terminal line characteristics such as baud rate. The file is in the form of a table. Each table entry is divided into five fields. These fields are listed in the following form:

label initial-flags final-flags login-prompt next-label

label — Identifies the gettydefs entry to getty . This could be a number or a letter. *label* corresponds to the line mode field in /etc/ttys.

initial-flags — Set the terminal line characteristics when getty first establishes the connection with the line. getty recognizes the flags described in termio (M) in the XENIX User and System Administrator Reference Manual. Sometimes, the only flag that appears in this field is the one that sets the baud rate. For example, B300 sets the speed to 300.

final-flags — Set the terminal line characteristics just before getty executes the login command. These flags describe the operating characteristics for the line. The baud flag (B) sets the baud rate again. Other common flags include SANE (a composite flag that sets a number of terminal characteristics to reasonable values), TAB3 (expands tabs with spaces), IXANY (enables any character to restart output), and HUPCL (hangs up line on final close). You can enter flags in any order.

login-prompt — Contains the login prompt message that greets users. The login message will be displayed exactly as you type it in this field, including spaces and tabs.

next-label — Identifies the next label in gettydefs for getty to try if the current one is not successful. getty tries the next label if a user types the **BREAK** key while attempting to log in to the system. Groups of entries, for example, for dial-up lines or for TTY lines, should form a closed set so that getty cycles back to the original entry if none of the entries is successful.

Each field is separated by pound sign (#), and each entry in gettydefs is separated by a blank line.

An entry in gettydefs might look like this:

```
1# B1200 PARENB OPOST ONLCR # B1200 SANE TAB3 HUPCL #Login: #2
```

The number 1 identifies this entry to getty. The next field selects a baud rate of 1200, parity-checking and post-processing of NL characters as the CR-NL character pair. The final settings in the third field include the baud rate (B1200), SANE (a composite flag for a number of characteristics), TAB3 (expands tabs to spaces) and HUPCL (hangs up line

on final close). The login prompt appears as "Login:", and if this setting is not successful, getty proceeds to label 2 in gettydefs.

CHANGING THE GETTYDEFS FILE

The file /etc/gettydefs already exists in your XENIX system, and has sets of entries for the operator's console, dial-up lines, and terminal lines. These different sets correspond to line mode settings in the /etc/ttys file. When the init program initializes the XENIX system, it passes line mode information to getty

You can edit gettydefs to add new terminal settings or to change existing ones. For example, the settings for terminal lines on your XENIX system might look like this:

```
C# B9600 PARENB CS7 OPOST ONLCR # B9600 SANE IXANY CS8 #Console Login: #C
7# B9600 PARENB CS7 OPOST ONLCR # B9600 SANE IXANY TAB3 #Login: #1
6# B9600 PARENB CS7 OPOST ONLCR # B9600 SANE IXANY #Login: #7
5# B4800 PARENB CS7 OPOST ONLCR # B4800 SANE IXANY #Login: #6
4# B2400 PARENB CS7 OPOST ONLCR # B2400 SANE IXANY #Login: #5
3# B1200 CS8 OPOST ONLCR # B1200 SANE IXANY #Login: #4
2# B300 CS7 OPOST ONLCR # B300 SANE IXANY #Login: #3
1# B300 CS8 OPOST ONLCR CR1 # B300 SANE IXANY #Login: #2
```

To change the gettydefs file so that there is an entry for a terminal running at 9600 baud that includes the CS8 flag, for 8-bit character size, do the following:

1. Log in as super-user.
2. Enter a text editor to edit the file gettydefs.
3. Add as a new second entry a line such as the following:

   B# B9600 PARENB CS8 OPOST ONLCR # B9600 SANE IXANY TAB3 #Login: #1

4. Change the final element of entry 7 from 1 to 8.

5. Exit the text editor, saving gettydefs.

The gettydefs file should now look like this:

```
C# B9600 PARENB CS7 OPOST ONLCR # B9600 SANE IXANY CSB #Console Login: #C
B# B9600 PARENB CSB OPOST ONLCR # B9600 SANE IXANY TAB3 #Login: #1
7# B9600 PARENB CS7 OPOST ONLCR # B9600 SANE IXANY TAB3 #Login: #8
6# B9600 PARENB CS7 OPOST ONLCR # B9600 SANE IXANY #Login: #7
5# B4800 PARENB CS7 OPOST ONLCR # B4800 SANE IXANY #Login: #6
4# B2400 PARENB CS7 OPOST ONLCR # B2400 SANE IXANY #Login: #5
3# B1200 CSB OPOST ONLCR # B1200 SANE IXANY #Login: #4
2# B300 CS7 OPOST ONLCR # B300 SANE IXANY #Login: #3
1# B300 CSB OPOST ONLCR CR1 # B300 SANE IXANY #Login: #2
```

Flags and permissible values for terminal settings are listed in termio (M) in the XENIX User and System Administrator Reference Manual.

When you add a new entry, be sure that the groups of entries in gettydefs form a closed set, so that the next-label field of the last entry directs getty back to the first entry in the group.

CHECKING THE TERMINAL SETTINGS

Each time you change the terminal line settings or add new entries to gettydefs, you should check to make sure that the new values are allowed by getty . To do this, use the getty command, the check option -c , and the filename.

For example, to check gettydefs, type:

```
getty -c /etc/gettydefs
```

If any of the values and settings in gettydefs are not permitted, getty -c will print these on your terminal screen.

For more information on getty and gettydefs, see getty(M) and gettydefs(F) in the XENIX Reference Manual.

REMOVING A TERMINAL

From time to time it may be necessary to remove a terminal from the system, for example, if you wish to replace it with some other device. Before you can remove a terminal, you must disable it with the disable command.

To remove a terminal, follow these steps:

1. Turn off the power to the terminal.

2. Log in as the super-user at another terminal.

3. Use the disable command to disable the terminal. The command has the form:

   disable *specialfile*

   where *specialfile* is the name of the serial line to which the terminal is attached. For example, the command:

   disable /dev/tty01

   disables the terminal connected to serial line /dev/tty01.

4. Disconnect the terminal from the system.

The serial line that was previously connected to the terminal is now free to accept another device.

When using the disable command, make sure that you wait a full minute between each use of the command. Failure to do so can cause a system crash.

CONFIGURING KEYBOARDS

You can attach any of 13 national language keyboards to the console and any of 7 to a WS584 terminal. You must identify to the XENIX system the nationality of your keyboards. For terminals, this is done automatically when you configure your system with a particular configuration file (see Chapter 9, "Using Configuration Files"). With the console keyboard, on the other hand, you must stipulate the nationality whenever you wish to use something other than the U.S. default.

With national version terminal keyboards, you have the option of simulating a U.S. keyboard.

You also have the option of assigning values to the function keys on the console keyboard.

SELECTING THE CONSOLE KEYBOARD LANGUAGE

To select a keyboard language for the console that is other than the U.S. default, use the select command. The syntax of this command is as follows:

```
select countrycode
```

where *country-code* is a two-letter abbreviation for the national version selected. It must be one of the following:

COUNTRY CODE	COUNTRY
DA	Denmark
FS	Finland-Sweden
FR	France
GR	German
IT	Italy
NO	Norway
PO	Portugal
SP	Spain
ES	Spain 2
SF	Swiss-French
SG	Swiss-German
UK	United Kingdom
US	United States

Note that the *country-code* must be entered in upper-case.

Follow the select invocation with the following command:

```
stty -istrip
```

This removes the 7-bit stripping mechanism, allowing the system to interpret keyboard input in the 8-bit character range employed by national language versions.

For example, to establish the console keyboard as Italian, type:

```
select IT
```

```
stty -istrip
```

The keyboard immediately starts behaving as an Italian keyboard.

There are certain restrictions on the use of a console whose keyboard is set to a national language. The following utilities are amongst those that will not recognise 8-bit characters and so should be avoided:

```
echo
edit
grep
vi
vsh
```

You should also avoid 8-bit characters when creating filenames.

To make a language selection permanent, you should include the appropriate select command in a .login or .profile file.

SWITCHING CONSOLE KEYBOARD LANGUAGES

If your console keyboard has been configured for a language version other than U.S., you can switch back and forth between the default and U.S. versions with simple key combinations.

To switch from a national keyboard to the U.S. version, press:

CTRL ALT F1

To switch back from a U.S. to a default national keyboard version, press:

CTRL ALT F2

SWITCHING TERMINAL KEYBOARD LANGUAGES

With terminal keyboards configured for a language version other than U.S., you can switch back and forth between the default and U.S. versions with simple key combinations. This is particularly useful in the light of the restrictions on 8-bit terminal input, which are the same as those for consoles noted above.

The following table shows the necessary key combinations for the six non-U.S. versions available:

DEFAULT NATIONAL VERSION	SWITCH TO U.S. VERSION BY PRESSING...	SWITCH BACK TO DEFAULT BY PRESSING...
France	CTRL -	CTRL /
Germany	CTRL -	CTRL /
Norway/Denmark	CTRL -	CTRL /
Spanish	CTRL -	CTRL /
Sweden/Finland	CTRL -	CTRL /
United Kingdom	CTRL /	CTRL /

ASSIGNING VALUES TO FUNCTION KEYS

To assign values to the function keys on the left of the console keyboard, use the setkey command. The syntax of this command is either:

```
setkey keynum string
```

or:

```
setkey keynum -d keyvalue
```

where *keynum* is a number from 1 to 10 representing the function key whose value you wish to set.

The *string* argument should be enclosed within double quotation marks. This allows you to include control characters such as CR within the string. For example, to set function key 1 so that it does an ls -l whenever it is pressed, type:

```
setkey 1 "ls -l
"
```

and press CR . Note the additional CR after the command string but before the closed quotes.

You can also assign control characters to function keys by including as the *string* argument the ^ character followed by the character that is 20 hex on in the ASCII tables from the control character itself. For example, to assign the linefeed character (0A hex) to function key 2, you would use the * character (2A hex) and so type:

```
setkey 2 "^*"
```

To assign any of the 256 8-bit ASCII characters to a function key, use the -d option followed by the decimal number of the character in the ASCII tables.

setkey assignments can be saved by putting them in a script file. For example, you could create a file called setkeys which included setkey statements for each function key. These statements would then be acted

upon when you typed:

```
sh setkeys
```

To establish setkey assignments on a permanent basis, include them within the /etc/rc file (for system-wide settings) or a .profile file (for user-specific settings).

ADDING A LINEPRINTER

This section explains how to add new lineprinters to your XENIX system and how to use the lineprinter commands to organize and control your printing.

The XENIX lineprinter spooling system is a collection of commands that help you, as system administrator, to efficiently install, monitor and control the lineprinters serving your system. When a user requests to print a file using the lp command, the lineprinter system responds with a request ID. This consists of the name of the printer the file will be printed on and a unique number identifying the file. With this request ID, the user can find out the status of the print request or cancel it. The lp options help the user control printer output. For more information on lp , see the XENIX User and System Administrator Reference Manual.

A few terms need to be defined first. A device is the target for lp output. It could be a hard-wired printer, a terminal that is sometimes used as a printer, or a regular file. A device can be represented by a full XENIX pathname. A printer is the name assigned by the system administrator to represent a device. This name can have up to 14 characters. At different times, a printer may be associated with different devices. Printers can be grouped together into classes which are ordered lists of printers. A destination can be a printer or a class.

Consult the hardware manuals provided for your computer and lineprinters for information on making the physical connection between your system and printing devices.

INSTALLING A PRINTER: lpinit

To install new printing devices on your XENIX system, use the lpinit command. Before you use lpinit you should first know the port to which the lineprinter is connected or the XENIX pathname of the device (for example, /dev/lp0) and the lineprinter interface program. A model interface program is supplied with your XENIX system. For more information on printer interface programs, see "Printer Interface Programs" later in this section.

When you invoke the lpinit command, you will be asked a series of questions for which the default answers are displayed. If you wish to choose the default answer, simply press CR . If you want to supply your own answers, type in the information as you are prompted. If you

make a mistake while responding to the questions, just press the DELETE key or the INTERRUPT key and start again.

The following example adds a lineprinter to your XENIX system. The printer name is printer1, the device pathname is /dev/lp0, and the printer will be the default printer for your system.

1. Type the command:

   ```
   /etc/lpinit
   ```

   lpinit displays the following message on your screen:

   ```
   How many printers do you want to connect to the system?

   Enter your choice (1, 2 or 3):
   ```

2. Type 1 and press CR if you wish to connect a single printer, type 2 and press CR to connect two, type 3 and press CR to connect three. In this example select 1 .

   The following message then appears on the screen:

```
The first printer is attached to a

1.  Primary Parallel Port on the Mother Board.

2.  Alternate Parallel Port on a Serial/Parallel Adapter.

3.  Parallel Port on a Monochrome and Printer Adapter.

4.  Serial Printer on the Serial Port on the Mother Board.

Enter one of the options above
(default = 1)                                    :
```

3. Select the appropriate option, then type the corresponding number and press CR . Note that, if you are connecting two or three printers, this prompt will appear twice or three times and you will have to identify two different connection points. In the current example just press CR to accept the default of the mother board.

   lpinit displays the following message:

```
Enter the name for the first printer
(default = linepr)                    :
```

4. Printer names may be up to 14 characters long and may be any combination of numbers, letters, or underscore characters. Type the printer name, printer0, and press CR .

   lpinit displays the following message:

```
Enter an interface program for printer linepr
(default = /usr/spool/lp/model/olipr15)     :
```

5. This example uses the default choice, /usr/spool/lp/model/olipr15, so simply press CR .

   Next, lpinit displays the following message:

```
/usr/lib/lpshut: scheduler not running
destination "printer0" now accepting requests
printer "printer0" now enabled
```

   This indicates that lpinit sought to temporarily shut down the lineprinter scheduling program, **lpsched** , but found that it had never been started (if it had been the message "scheduler stopped" would have appeared instead). It then automatically gave the instructions to enable printer0 to accept print requests.

6. After enabling printer0, lpinit asks if this printer will be the system default printer:

```
Is printer printer0 the default printer ? [y/n]
(default = y)                                    :
```

7. You can type y (for "yes") or n (for "no") depending upon whether you want users' print requests to be automatically routed to printer0 or not. Press CR to accept the default of "yes".

   If you had elected to configure more than one printer, **lpinit** would have cycled through the previous prompts the appropriate number of times, then asked you which, if any, of the named printers you wanted as the default.

After you have responded to these questions, lpinit restarts lpsched and users can begin printing files on the new printer.

You can also add printers to your system using the lpadmin command discussed later in this section; however, you will need to give separate commands to stop lpsched , to enable the printer, and to allow it to accept print requests. For more information on these programs and commands, see the remainder of this section.

STOPPING THE PRINT SPOOLING DAEMON: lpsched

A daemon is a process that runs in the background, independent of a terminal, and performs a function. The spooling daemon, lpsched , routes print requests through the correct printer interface program and then to the lineprinter. You cannot print on your system unless lpsched is running. The lpsched program starts automatically each time you restart your XENIX system. Sometimes, it is necessary to stop lpsched , especially if you want to reconfigure printers or if you want to add new printers using the lpadmin command (lpinit automatically stops and restarts lpsched for you).

This section explains how to find out whether or not lpsched is running, how to stop and restart it, and how to recreate it if necessary.

To find out whether or not lpsched is running, type:

```
lpstat -r
```

In response to the lpstat command, your screen displays a message that lpsched is either running or not running.

To shut down the scheduler, lpsched , type:

```
/usr/lib/lpshut
```

lpsched stops running and all printing stops as well. Printing requests stopped in the middle of printing reprint when lpsched starts again.

After you have finished configuring the printers, you should restart lpsched . To do this, type:

```
/usr/lib/lpsched
```

Next, check lpstat -r to confirm that lpsched is running. If it is not, type the following on two lines:

```
rm -f /usr/spool/SCHEDLOCK
/usr/lib/lpsched
```

This removes any existing version of SCHEDLOCK, which is a routine that ensures that only one version of lpsched is running at a time, then reinvokes lpsched .

Each time lpsched sends e print request to an interface program, it records an entry in a log file, /usr/spool/lp/log. The entry includes the user name, the request ID, the name of the printer the request will be printed on, and the date and time requested. lpsched also records any error messages in this file. After you have stopped lpsched , the log file is renamed /usr/spool/lp/oldlog and lpsched starts a new log file. Requests waiting to be printed before lpsched was stopped may have an entry in both log files.

For more information on lpsched , see lpsched(C) in the XENIX User and System Administrator Reference Manual.

CHANGING PRINTER CONFIGURATION: lpadmin

After you have added printers to your system using lpinit , you may want to alter their configuration. For example, you may want to modify an interface program or create a new class of printers. To change printer configuration, use the lpadmin command with the -p option. If lpsched , the printing scheduler, is running, lpadmin will not alter the configuration except where noted. To stop lpsched , see the previous subsection.

The syntax for lpadmin -p is as follows:

```
/usr/lib/lpadmin -p printer options
```

where *printer* is the name of the printer to be reconfigured and *options* include the following:

-c Creates a class of printers or adds a printer to a class. The same rules apply to class names as to printer names: up to 14 alphanumeric characters or underscores. Print requests sent to a class of printers will be printed on the first available member of the class.

For example, to add a printer named laser to a class of printers named class1, type:

```
/usr/lib/lpadmin -plaser -cclass1
```

-r Removes printer from a class of printers. If the printer is the last member of the class, lpadmin removes the class as well. For example, to remove laser from class1, type:

```
/usr/lib/lpadmin -plaser -rclass1
```

-i Assigns a new interface program to printer. The pathname of the interface program must follow -i . For example, to establish /usr/spool/lp/model/dumb as the interface program for a printer named prt1, type:

```
/usr/lib/lpadmin -pprt1 -i/usr/spool/lp/model/dumb
```

You can also use the lpadmin command to add new printers to your system. Unlike the lpinit command, lpadmin only configures the new printer. You must separately perform the steps to stop lpsched , enable the new printer, and give the accept command so that the printer will accept print requests. For more information on using lpadmin to add new printers to your system, see in the XENIX User and System Administrator Reference Manual.

CREATING A DEFAULT PRINTER

You can specify one printer or one class of printers on your system to be the default destination of a print request. All files will be sent to the system default destination for printing unless the user specifies a printer when using the lp command. For more information on lp and its options, see the XENIX User and System Administrator Reference Manual.

To create or change the system default destination, use lpadmin and the -d option. You can change this option even when the scheduler, lpsched , is running.

For example, to make a class of printers called classb the system default destination, type:

```
/usr/lib/lpadmin -dclassb
```

To establish no default destination, you use the -d option without a printer name:

```
/usr/lib/lpadmin -d
```

For more information, see **lpadmin(C)** in the XENIX User and System Administrator Reference Manual.

REMOVING PRINTERS

You can remove printing destinations, either printers or classes of printers, if there are no print requests routed to them. If you remove a class of printers, the individual printers remain in the system. Removal of the last remaining printer in a class of printers removes the class as well. To remove a printer or class of printers, use lpadmin and the -x option. For example, to remove a printer called printer2 from your system, type:

```
/usr/lib/lpadmin -xprinter2
```

If you wish to temporarily remove a printer because of a breakdown, see "Enabling and Disabling Printers" later in this section. If you wish to temporarily reroute print requests to other printers, see the next two subsections.

For more information on lpadmin and its options, see **lpadmin(C)** in the XENIX User and System Administrator Reference Manual.

MOVING REQUESTS BETWEEN PRINTERS: lpmove

You can move print requests between printing destinations by using the lpmove command. This command will not move print requests while the scheduler, lpsched , is running. To stop lpsched , see "Stopping the Print Spooling Daemon: lpsched" earlier in this section. lpmove will move individual print requests by request ID, or all requests waiting to be printed on a particular printer.

For example, to move a request with a request ID of quick-532 to a printer named slow, type:

```
/usr/lib/lpmove quick-532 slow
```

The print request now has a new request ID: slow-532.

To move all requests on a printer named quick to slow, type:

```
/usr/lib/lpmove quick slow
```

For more information on lpmove , see lpsched(C) in the XENIX User an System Administrator Reference Manual.

ACCEPTING AND REJECTING PRINT REQUESTS: accept

The accept command allows printers or classes of printers to accept print requests made with the lp command. After a printer has been properly configured, you can allow it to accept print requests; however, the printer will not begin printing the requests until the enable command is given. If you added a printer to your system using the lpinit command, these steps were automatically taken. For information on enable , see the next subsection.

For example, to have print requests accepted for a class of printers named class1, type:

```
/usr/lib/accept class1
```

If you want to prevent requests from being routed to a printer, you can use the reject command. The -r option allows you to send users a message explaining why a printer is out of service.

For example, to prevent printing requests from being routed to a printer called printer4 because of repairs, type:

```
/usr/lib/reject -r "printer4 needs repair" printer4
```

A user who requests to print a file on printer4 will receive the following message:

```
lp:can not accept requests for destination "printer4"
--printer printer4 needs repair
```

To find out the acceptance status of printing destinations, type:

```
lpstat -a
```

For more information on lpstat , see lpstat(C) in the XENIX User and System Administrator Reference Manual. For more information on accept and reject , see accept(C) in the XENIX User and System Administrator Reference Manual.

ENABLING AND DISABLING PRINTERS

The enable command allows lpsched to print files on printers. A printer can accept requests for printing after you give the accept command for it, but in order to print the files, you must give the enable command as well.

For example, to enable a printer named daisy, type:

```
enable daisy
```

You can disable printers with the disable command. The scheduler, lpsched , will not send printing requests to disabled printers regardless of their acceptance status. The -r option allows you to send a message to users explaining why a printer has been disabled.

For example, to disable a printer named laser because of a paper jam, type:

```
disable -r "paper jam" laser
```

Users requesting the status of laser with the command "lpstat -plaser" will receive the following mesage:

```
printer "laser" disabled since Dec 5 10:15
paper jam
```

For more information on these two commands, see enable(C) and disable(C) in the XENIX User and System Administrator Reference Manual.

PRINTER INTERFACE PROGRAMS

Each printer on your system must have a printer interface program. This can be a shell script, C program, or any other executable program. Your XENIX system provides a model interface program. It is written as a shell script and can be found in /usr/spool/lp/model. You can use this program as is, modify it, or write your own interface program.

If you want to write or modify a printer interface program, the following information may be helpful.

At the time lpsched routes a printing request to a printer p, the interface program for p is invoked in /usr/spool/lp in the following form:

```
interface/P id user title copies options file
```

id Identifies the request id returned by lp .

user Gives the login name of the user who made the request.

title Gives an optional title chosen by the user.

copies Gives the number of copies requested.

options Lists printer dependent options. Each option should be separated by a blank space.

file Gives the full pathname of a file to be printed.

When the interface program is started, its standard input comes from the /dev/null file and both standard output and standard error output are directed to the printer's device. Devices are opened for reading, as well as writing, when file modes permit. In the case where a device is a regular file, all output is appended to the end of the file.

You may format your output through interface programs in any way you choose; however, you must ensure proper stty modes (terminal characteristics such as baud rate and output options). In a shell script interface, this means that the printer's device must be open for reading, that is, it must take the standard input for the stty command from the device.

The file /etc/default/lpd contains a line "BANNERS= *d* "" where *d* is the number of banner pages you want to print at the front of every printing request. Interface programs should examine this file and behave accordingly.

After printing is completed, the interface program should exit with a code showing that the print job was successful. Exit codes are interpreted by the printer scheduler, lpsched , as follows:

EXIT CODE	MEANING TO lpsched
0	Print job was successful.
1 to 127	**lpsched** found a problem while printing this particular request, for example, too many unprintable characters. This problem will not affect future printing requests. **lpsched** notifies users by mail that there was an error in printing the request.
greater than 127	These codes are reserved for internal use by **lpsched**. Interface programs must not exit with codes in this range.

Finally, when problems occur in printing that are likely to affect future printing requests, the printer interface program should disable printers so that print requests are not lost. When a busy printer is disabled,

the interface program retains the print requests so that they are not lost.

For more information on printer interface programs, see lpadmin(C) in the XENIX User and System Administrator Reference Manual.

9. USING CONFIGURATION FILES

ABOUT THIS CHAPTER

This chapter explains how to use configuration files to install device drivers.

CONTENTS

INTRODUCTION

This chapter explains how to configure your XENIX kernel in such a way that it recognises the particular arrangement of hardware in your system. It describes how configuration files are used to define your system and the XENIX utilities you need to use to install them.

WHAT IS A CONFIGURATION FILE?

For each type of peripheral device in a XENIX system, you must have incorporated into the XENIX kernel an appropriate device driver. A XENIX device driver is a set of routines that communicates with a hardware device to control and perform input/output (I/O) operations. It is the software interface between the system and the device. Installing a device driver is as essential to the operation of the device as the actual hardware installation.

A configuration file is a file which groups device drivers together in order to accommodate a complete hardware configuration. So, for example, one configuration file will contain device drivers for a multi-user system with diskette drive and parallel printer support, while another will support a configuration that has all these features plus a cartridge tape drive.

There is a separate set of configuration files for each national version configuration that includes national version WS584 terminals.

A configuration file appropriate to your configuration should be invoked at or soon after installation time (a default kernel is supplied on your distribution so that you can get your XENIX system up and running without first incorporating a configuration file). But if you, for example, add a cartridge tape drive at a later date or choose to employ terminal keyboards of a different nationality, you will need to reconfigure your XENIX kernel with a different set of device drivers.

Your XENIX distribution contains a set of configuration files that cater for all the most likely, recommended hardware configurations. However, you can also create configuration files of your own if you have a non-standard configuration. The last section of this chapter explains how to do this.

Note that each configuration file should define a complete hardware configuration, in order that it can be incorporated within the original linking kernel supplied in your distribution. You should not attempt to add in partial configuration files to already-configured kernels.

BEFORE YOU START

Before you install a configuration file on your XENIX system, you should:

- Install the hardware device on your computer system according to the manufacturer's instructions.
- Run SETUP from your CUSTOMER TEST diskette to indicate the new hardware configuration.
- Identify the configuration file appropriate to your particular hardware and nationality. A complete list of the supplied configuration files is contained in Appendix C.

The following section describes the procedure for incorporating an existing configuration file into your XENIX kernel. If you need to create a new configuration file, see the section after that.

INSTALLING A CONFIGURATION FILE

Installing a configuration file involves the following two steps:

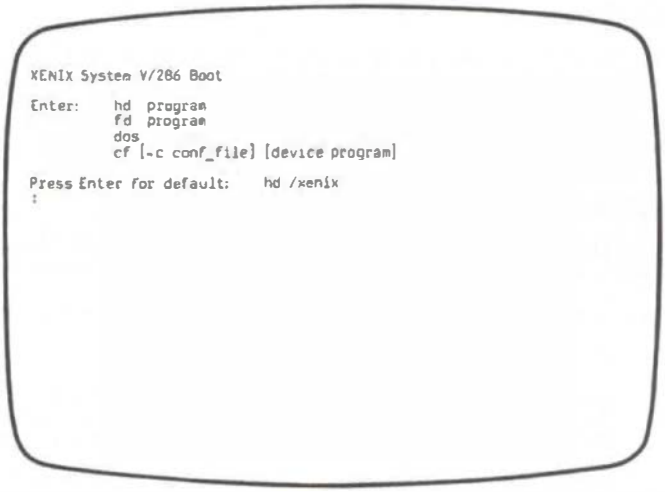
1. Loading a configuration file
2. Creating a new XENIX kernel

They should be performed one after the other. The complete procedure need only be performed once, for any one hardware configuration.

LOADING A CONFIGURATION FILE

This subsection describes how to load the device drivers contained in a configuration file so that they can be incorporated within the XENIX kernel.

To begin the loading procedure, you must reboot the system. When you boot the system, the screen display looks like this:



Instead of pressing CR to accept the default and boot up the existing XENIX from the hard disk, you must respond to the prompt ":" with some form of the cf command.

The syntax of the cf command, as indicated in the boot display, is as follows:

cf [-c *conf_file*] [*device program*]

where *conf_file* is the name of the configuration file you wish to incorporate, and *device* and *program* identify the destination device for the file and the destination linking kernel.

The default for *conf_file* is config.sys. However, you should only use this default if you have first renamed your chosen configuration file config.sys. Otherwise you should type the actual name of the configuration file (see Appendix C for a list of the Olivetti-supplied files).

The defaults for *device* and *program* are hd and /lib/sys/xenix, and unless you have moved the linking XENIX kernel from its standard position in the /lib/sys directory on the hard disk, or renamed it, you should accept these defaults. (The *device* parameter may be either hd (hard disk) or fd (diskette) and the *program* parameter should specify the complete pathname of whatever file is to be used as the linking kernel.)

For example, to load the configuration file /lib/sys/config11.sys using the default device and kernel, you would type:

cf -c /lib/sys/config11.sys

and press CR .

When the cf command has completed its loading task the boot display returns to the screen. You should now press CTRL D to boot up XENIX (as yet not reconfigured) from the hard disk and enter maintenance mode. When you have done so you should proceed to the second step, described below.

CREATING A NEW XENIX KERNEL

This subsection describes how to incorporate within the XENIX kernel the device drivers loaded by a previous invocation of the cf command.

To create a new XENIX kernel, go through the following steps:

1. Log in as super-user.

2. Rename the currently-used kernel, /xenix. Give it a name such as /xenixold that readily identifies it. This procedure ensures that you do not overwrite a kernel that you may want to use again.

3. Invoke the xconf command. The syntax of the xconf command is as follows:

   xconf -o *kernel_file* -c *conf_file device program*

   where *kernel_file* is the name of the new kernel to be generated, and *conf_file* , *device* and *program* the configuration file, destination device and linking kernel identified in the previous cf command invocation.

   The *kernel_file* should generally be given as /xenix. Otherwise, neither the ps nor the pstat command will produce correct output when you execute it using the new kernel. (Nor will the default boot response hd /xenix be valid any longer.)

For example, to create a new kernel using the config11.sys file loaded previously, type:

xconf -o /xenix -c /lib/sys/config11.sys hd /lib/sys/xenix

and press CR .

Note that the old kernel is still in control of your system at the conclusion of the xconf procedure. The new kernel will only become active after a subsequent system shutdown and reboot.

CREATING A CONFIGURATION FILE

If you wish to add a peripheral device that is not catered for in the standard range of configuration files supplied with your XENIX operating system, you must create a new configuration file. This must contain the installable device driver for the peripheral, along with drivers for all the other types of device in the configuration.

You will normally be supplied with an installable device driver on diskette when you acquire a non-standard peripheral device. You can also write your own device driver (see the XENIX V System and Application Software Development Tools User Guide for details).

Creating and installing a new configuration file involves the following five steps:

1. Installing the device driver software modules in the system.
2. Modifying a configuration file to include the new drivers.
3. Creating special device files for each new device.
4. Loading the modified configuration file.
5. Creating a new version of the XENIX kernel.

The last two steps are as described in the previous section. The first three are described in this section.

This section gives two examples of non-standard peripherals and their device drivers -- one a high-quality printer and the other a high-capacity diskette drive. Note that these are only examples and do not identify actual devices.

INSTALLING DEVICE DRIVER MODULES

To begin the installation of the new device drivers, you will usually copy the device driver software modules from the diskette that is supplied with the device to the /lib/sys directory. However, if the instructions provided by the manufacturer of the device conflict with the instructions presented here, the manufacturer's instructions take precedence.

Follow the instructions given below to install the device driver modules:

1. Check the contents of the diskette(s) provided with the devices. To do this, insert each diskette in turn in drive 0 and type:

   ```
   tar tf /dev/rfd0
   ```

   The diskette provided with the high-quality printer will typically display the following contents:

   ```
   /lib/sys/config.sys
   ```

```
/lib/sys/hqp.x
```

The diskette provided with the high-capacity diskette drive will typically display the following contents:

```
/lib/sys/config.sys
/lib/sys/hcd.x
```

2. Check the current contents of the /lib/sys directory to make sure you will not overwrite an existing file by installing the device driver files. Typing the command line:

```
lc /lib/sys
```

will display the contents of the directory. If there is an existing file of the same name as one you wish to install, you must rename the existing file in order to preserve its contents. Let us assume that you already have a config.sys file in your /lib/sys directory.

3. Rename the existing config.sys file in the /lib/sys directory to Mconfig.sys to preserve its contents when the device driver files are installed. To do this, type:

```
cd /lib/sys
mv config.sys Mconfig.sys
```

4. Insert in drive 0 the diskette that contains the device driver software for the printer. Copy the files from this diskette on to the system by typing:

```
tar xf /dev/rfd0
```

5. Rename the printer's config.sys file to config.hqp to preserve its contents. To do this, type:

```
mv config.sys config.hqp
```

6. Insert in drive 0 the diskette that contains the device driver software for the diskette drive. Copy the files from this diskette on to the system by typing:

```
tar xf /dev/rfd0
```

7. Rename the diskette drive's config.sys file to config.hcd to preserve its contents. To do this, type:

```
mv config.sys config.hcd
```

8. Restore the system's original config.sys file to its original name by typing the command line:

```
mv Mconfig.sys config.sys
```

You are now ready to modify the system configuration file to include the new device drivers.

MODIFYING A CONFIGURATION FILE

The next step in installing the device drivers is to edit a configuration file in order to include the new device drivers. This example assumes that you have made a copy of the configuration file that identifies your current hardware configuration and given it the default name config.sys. While it is not necessary to use this default name, you should make a copy of the file you are about to edit, so as to preserve the original contents in case of mistakes.

To make a copy of the configuration file config11.sys and rename it config.sys, type:

```
cp config11.sys config.sys
```

and press CR .

The device specifications you need to include in the configuration file should be supplied to you with the installation instructions for the device. Before you begin editing a configuration file, you should have a general understanding of how device drivers are specified.

The lines in a configuration file are either of two types: a comment line, which begins with a (#) character and allows you to include some descriptive information, or an installable device driver specification, which begins with the characters ``DD''. Each device driver specification line contains an additional seven fields. These fields contain the following information:

- The name of the device. This is a mnemonic name associated with the device that is being installed. It is important that this name be used exactly as specified in the installation instructions, since the device driver software uses this name.

- The command which specifies whether the device you are installing will be added to the current configuration, will replace a previously existing device, or will share a device interrupt level with another device. Interrupt levels are described further below.

- The major device number of this device, if it is a block device, or a -1, if it is a character device only.

- The major device number of this device, if it is a character device, or a -1, if it is a block device only.

- A reserved field which should always contain an asterisk (*) character.

- The list of interrupt vector numbers that are associated with this device. This list is separated by commas and contains numbers in the range zero to seventy one. If the device is a non-interrupting type, this field should contain a -1.

- The full pathname of the file that contains the device driver module.

In most cases, you simply copy the device driver specification line from the config.sys file that is supplied with the device into the copy of the configuration file that currently describes your system. You must, however, be careful to check for clashes between existing devices and new devices. These clashes can occur in any of three fields in a device driver specification, the major device number for either the block or character device fields, and the interrupt vector numbers used by the device.

The rules for resolving these clashes are fairly simple:

- If an existing device has the same major device number as that given for the new device, choose a new, unique number for the new device and use that as the major device number. This rule applies to both the block and character device fields. Note that if you have to change the major device number from that supplied in your installation instructions, you must use this new number when you create the special device files described in the next subsection. If you wish to replace an existing device with the new device, you should choose the same major device number as the existing device and specify the REP option in the command field.

- If the interrupt vector number for the new device is different than all others listed in the existing configation file, put the string ADD in the device specification line in the command field and set the interrupt vector field as shown in the supplied instructions.

- If the interrupt vector number is the same as in an existing device specification, put the string SHR in both the existing and new device specification descriptions in the command field. The interrupt vector numbers should be the same for both devices.

Follow the instructions below to modify the config.sys file to include the new printer and diskette drive:

1. Examine the contents of the printer configuration file config.hqp and the diskette drive configuration file config.hcd . To do this, type:

   ```
   cat  confin.hqp  config.hcd
   ```

   For purposes of continuing with our example, we will assume the contents of config.hqp to be:

   ```
   #Whizzo Computer Products High Quality Printer Driver
   #
   DD      hqp     ADD     -1      7 * 5           /lib/sys/hqp.x
   ```

   For purposes of continuing with our example, we will assume the contents of config.hcd to be:

   ```
   #Whizzo Computer Products High Capacity Disk Drive Driver
   #
   DD      hcd     ADD     2       2 * 1           /lib/sys/hcd.x
   ```

2. Examine the contents of the current config.sys file by typing the command line:

   ```
   cat config.sys
   ```

   The current config.sys file might look like this:

   ```
   #DD name    cmd    bdev   cdev   line   ivec   path
   DD  fd      ADD    2      2      *      6      /lib/sys/fdidd.x
   DD  td      ADD    -1     5      *      3,4    /lib/sys/tdidd.x
   DD  lp      ADD    -1     6      *      5,7    /lib/sys/lpidd.x
   ```

3. Resolve the major device number conflict between the new hcd driver and the existing fd driver by changing both major device numbers for the hcd device to one. The corrected device driver specification for hcd will be:

   ```
   DD  hcd     ADD    1      1      *      1      /lib/sys/hcd.x
   ```

4. Resolve the interrupt level number conflict between the new hqp driver and the existing lp driver by specifying the SHR string so that both drivers can share interrupt level five. The corrected device driver specifications for the drivers will be:

   ```
   DD  lp      SHR    -1     6      *      5,7    /lib/sys/lpidd.x
   DD  hqp     SHR    -1     7      *      5      /lib/sys/hqp.x
   ```

5. To conclude, the modified system configuration file config.sys that incorporates the new drivers should look like:

   ```
   #DD name    cmd    bdev   cdev   line   ivec   path
   DD  fd      ADD    2      2      *      6      /lib/sys/fdidd.x
   DD  td      ADD    -1     5      *      3,4    /lib/sys/tdidd.x
   DD  lp      SHR    -1     6      *      5,7    /lib/sys/lpidd.x
   DD  hcd     ADD    1      1      *      1      /lib/sys/hcd.x
   DD  hqp     SHR    -1     7      *      5      /lib/sys/hqp.x
   ```

You have completed the process of modifying the system configuration file to include the new devices.

CREATING SPECIAL DEVICE FILES

A file must exist in the file system for each newly installed device so that programs can access the new devices. These files, called special device files, are usually located in the /dev directory. The installation instructions supplied with the device will specify the name you should use for the special device file and the other parameters associated with it.

Special device files are created using the mknod command with a set of required parameters. You must supply the name of the special device file, the device type (block or character), and the major and minor device numbers associated with the device.

To create special device files for the two new devices in our example, follow the instructions below:

1. Move to the /dev directory where you will create the special device files. To do this, type:

   ```
   cd /dev
   ```

2. Create the special device file for the diskette drive as a block device. To do this, type the command line:

   ```
   /etc/mknod hcd0 b 1 0
   ```

   where "hcd0" is the name of the special device file, "b" specifies that the device is a block device, "1" is the major device number, and "0" is the minor device number. Note that we used the modified device numbers for the hcd driver rather than the ones originally supplied on the diskette.

3. Create the special device file for the diskette drive as a character device. To do this type the command line:

   ```
   /etc/mknod rhcd0 c 1 0
   ```

   Note that an "r" has been prepended to the special device file name to indicate that this is the character special ("raw") form of the device.

4. Create the special device file for the high speed printer. This device is only a character device, so only one special device file needs to be created for it. To create the file, type the command line:

   ```
   /etc/mknod hqp c 7 0
   ```

You have finished creating the necessary special device files for your two new devices.

ABOUT THIS CHAPTER

This chapter explains how to solve system problems.

CONTENTS

INTRODUCTION

This chapter explains how to solve problems that affect the operation of the system. The problems range in complexity from how to fix a nonechoing terminal to how to restore lost system files.

RESTORING A NONECHOING TERMINAL

A nonechoing terminal is any terminal that does not display characters typed at the keyboard. This abnormal operation can occur whenever a program stops prematurely as result of an error, or when the user presses the **BREAK** key.

To restore the terminal to normal operation, follow these steps:

1. Press the **CTRL J** key sequence. The system may display an error message. If it does, ignore the message.

2. Type:

   ```
   stty sane
   ```

   The terminal does not display what you type, so type carefully.

3. Press **CTRL J** .

After pressing the **CTRL J** key sequence, the terminal should be restored and you may continue your work.

SOLVING LINEPRINTER PROBLEMS

No printing can be done on the lineprinter spooling system unless the print scheduler, lpsched , is running. To check the status of lpsched , type:

```
lpstat -r
```

To restart lpsched , type the following on two lines:

```
rm -f /usr/spool/lp/SCHEDLOCK
/usr/lib/lpsched
```

The lock file /usr/spool/lp/SCHEDLOCK prevents more than one version of lpsched from running at any one time.

Access to files and directories in /usr/spool/lp by lp can be another source of spooling problems. You can check the lpsched log file, /usr/spool/lp/log . This is a record of the print scheduler's activity and errors. If lpsched refuses to run or a printer refuses to print, check to make sure that:

- The printer is enabled; see lp(C) in the XENIX User and System Administrator Reference Manual.

- The files and directories in /usr/spool/lp are readable and writable by lp .

For more information on the lineprinter spooling system, see "Adding a Lineprinter" in Chapter 8.

STOPPING A RUNAWAY PROCESS

A runaway process is a program that cannot be stopped from the terminal at which it was invoked. This occurs whenever an error in the program "locks up" the terminal, that is, prevents anything you type from reaching the system.

To stop a runaway process in a multi-terminal system, follow these steps:

1. Go to a terminal that is not locked up.

2. Log in as the super-user.

3. Type:

   ```
   ps -ef
   ```

   and press CR . The system displays all current processes and their process identification numbers (PIDs). Find the PID of the runaway program.

4. Type:

   ```
   kill PID
   ```

   and press CR . The PID is the process identification number of the runaway program. The program should stop in a few seconds. If the process does not stop, type:

   ```
   kill -9 PID
   ```

   and press CR

The last step is sure to stop the process, but may leave temporary files or a nonechoing terminal. To restore the terminal to normal operation, follow the instructions in "Restoring a Nonechoing Terminal" earlier in this chapter.

To stop a runaway process in a single-terminal system, you must turn off the power to the system. You will need to restart the system by following the instructions in Chapter 3, "Starting and Stopping the System".

REPLACING A FORGOTTEN PASSWORD

The XENIX operating system does not provide a way to decipher an existing password. If a user forgets a password, the system administrator must change the password to a new one. To change an ordinary user password, follow the instructions in Chapter 4, "Preparing XENIX for Users".

REMOVING HIDDEN FILES

A hidden file is any file whose name begins with a period (.). You can list the hidden files in a directory by typing:

```
lc -a
```

and pressing CR .

You can remove most hidden files from a directory by typing:

```
rm .[a-z]*
```

and pressing CR . Remaining files can be removed individually with the rm command.

RESTORING FREE SPACE

The system displays an "out of space" message whenever the root directory has little or no space left to work. To restore system operation, you must delete one or more files from the root directory. To delete files, follow the steps outlined in Chapter 6, "Maintaining File Systems".

RESTORING LOST SYSTEM FILES

If a system program or data file is accidentally modified or removed from the file system, you can recover the file from the periodic backup disk with the sysadmin program. To restore the files, follow the instructions in Chapter 7, "Backing Up File Systems".

RESTORING AN INOPERABLE SYSTEM

On very rare occasions, one or more of the critical XENIX system files may be accidentally modified or removed, preventing the system from operating. In such a case, you must reinstall the XENIX system and restore user program and data files from backup disks. To reinstall the system, follow the instructions in the first part of this guide. To restore files from backup disks, follow the instructions in Chapter 7, "Backing Up File Systems".

RECOVERING FROM A SYSTEM CRASH

A system crash is a sudden and dramatic disruption of system operation that stops all work on the computer. System crashes occur very rarely. They are usually the result of hardware errors or damage to the root file system which the operating system cannot correct by itself. When a system crash occurs, the system usually displays a message explaining the cause of the error, then stops. This gives the system administrator the chance to recover from the crash by correcting the error (if possible) and restarting the system.

A system crash has occurred if 1) the system has displayed at the system console a message beginning with "panic:", or 2) the system refuses to process all input (including INTERRUPT and QUIT keys) from the system console and all other terminals.

To recover from a system crash, follow these steps:

1. Use the error message(s) displayed on the system console to determine the error that caused the crash. If there is no message, skip to step 3.

2. Correct the error, if possible. A complete list of error messages and descriptions for correcting the errors is given in messages(M) in the XENIX User and System Administrator Reference Manual. (Even if the problem cannot be located or corrected, it is generally worthwhile to try to restart the system at least once by completing the remaining steps in this procedure.)

3. Turn off the computer and follow the steps described in Chapter 3, "Starting the System", to restart the system.

4. If the system will not restart, or crashes each time it is started, the operating system is inoperable and must be reinstalled. Follow the procedures described in the first part of this guide to reinstall the system and in Chapter 7, "Backing Up File Systems," to restore user's files.

5. If the system cannot be started from the Installation diskette in the distribution set, the computer has a serious hardware malfunction. Contact a hardware service representative for help.

CHANGING XENIX INITIALIZATION

One common problem is how to adapt the system initialization to suit your system environment. This problem occurs whenever you have added new devices such as terminals or disk drives to the system, and wish these devices to be automatically enabled or mounted whenever you start normal system operation. You can adapt system initialization by modifying the system initialization files.

The XENIX initialization files contain XENIX commands and/or data which the system reads at system startup or whenever a user logs in. The files typically mount file systems, start programs, and set home directories

and terminal types. The initialization files are named /etc/rc, .profile, /etc/profile , /usr/lib/mkuser/mkuser.prof and /etc/motd.

The system administrator may modify these files to create any desired initial environment. The files are ordinary text files and may be modified using a text editor such as ed (see the XENIX User Guide). Note, however, that the /etc/rc and .profile files contain XENIX commands and comments, and have the command file format described in the XENIX User Guide.

CHANGING THE /etc/rc FILE

The /etc/rc file contains system initialization commands. The system executes the commands at system startup. The commands display a startup message, start various system daemons, and mount file systems. You can display the contents of the file with the more command. Type:

```
more /etc/rc
```

and press CR .

You may change the contents of the file so that the system executes any set of commands you wish. For example, if you want the system to automatically mount a new file system, simply append the appropriate mount command in the file. The system will execute the command on each startup.

To append a command to the file, follow these steps:

1. Log in as the super-user.
2. Invoke a text editor and specify the /etc/rc as the file to be edited.
3. Locate the place in the file you wish to insert the command (e.g., if the command mounts a file system, insert it with other mounting commands).
4. Insert the command on a new line. Make sure you type the command correctly. The system will reject any incorrect command and all following commands when it reads the file at system startup.
5. Exit the editor.

No other changes to the file are required. Be careful not to delete any commands already in the file unless you are sure they are not needed.

CHANGING THE .profile, /etc/profile, AND /usr/lib/mkuser/mkuser.prof FILES

The .profile, /etc/profile and /usr/lib/mkuser/mkuser.prof files contain commands that initialize the environment for each user. The commands in these files are executed whenever the user logs in. The files usually contain commands that set and export various system variables (e.g., TERM, PATH, MAIL). These variables give the system information such as what terminal type is being used, where to look for programs the user runs, where to look for the user's mailbox, what keys to expect for the "kill" and "backspace" functions, and so on (see the XENIX User Guide).

There is one .profile file for each user account on the system. The files are placed in the user's home directory when the account is created. Ordinary users may modify their own .profile files, or allow the system administrator to make modifications. In either case, the file can be edited the same way as the /etc/rc file, using a text editor. Commands can be added or removed as desired.

CHANGING THE /etc/motd FILE

The message-of-the-day file, /etc/motd, contains the greeting displayed whenever a user logs in. Initially, this file contains the name and version number of the XENIX system. It can be modified to include such messages as a reminder to clean up directories, a notice of the next periodic backup, and so on.

The /etc/motd file is an ordinary text file, so you can change the message by editing the file with a text editor. One common change is to include a reminder to delete unused files in order to preserve disk space. In general, you should limit the size of the file to include no more than a screenful of information.

11. BUIDLING A MICNET NETWORK

ABOUT THIS CHAPTER

This chapter explains how to build a Micnet network.

CONTENTS

INTRODUCTION

A Micnet network allows communication between two or more independent XENIX systems. The network consists of computers connected by serial communication lines (RS-232 ports connected by cable). Each computer in the network runs as an independent system, but allows users to communicate with the other computers in the network through the mail , rcp and remote commands. These commands pass information such as mail, files and even other commands from one computer to another.

It is the system administrator's task to build and maintain a Micnet network. The system administrator decides how to connect the computers, makes the actual physical connections, then uses the netutil program to define and start the network.

This chapter explains how to plan a network and then build it with the netutil program. In particular, it describes:

- How to choose machine names and aliases
- How to draw the network topology map
- How to assign serial lines
- How to create the Micnet files
- How to distribute the Micnet files
- How to test the Micnet network

PLANNING A NETWORK

To build a Micnet network, you must give the netutil program the names of the computers that will be in the network, a description of how you will connect the computers, a list of the serial lines you will use, the names of the users who will use the network, and what aliases (if any) they will be known by.

To keep the task as simple as possible, you should take some time to plan the network and make lists of the information you will be required to supply. To help you make these lists, the following sections suggest ways to plan a network.

CHOOSING MACHINE NAMES

A Micnet network requires that each computer in the network have a unique machine name. A machine name helps distinguish each computer from other computers in the network. It is best to choose machine names as the first step in planning the network. This prevents confusion later on when you build the network with the netutil program.

A machine name should suggest the location of the computer or the people who are users on the computer; however, you may use any name you wish.

The name must be unique and consist of letters and digits. The Micnet programs use only the first eight characters of each name, so be sure those characters are unique.

The netutil program saves the machine name of a computer in a /etc/systemid file. One file is created for each computer. After you have built and installed the network, you can find out the machine name of the computer you are using by displaying the contents of this file.

CHOOSING A NETWORK TOPOLOGY

The network topology is a description of how the computers in the network are connected. In any Micnet network, there are two general topologies from which all topologies can be constructed. These are star and serial.

In a star topology, all computers are directly connected to a central computer. All communication passes through the central computer to the desired destination.

In a serial topology, the computers form a chain, with each computer directly connected to no more than two others. All communication passes down the chain to the desired destination.

A network may be strictly star, strictly serial, or a combination of star and serial topologies. The only restriction is that no network may form a ring. For example, you cannot close up a serial network by connecting the two computers at each end.

The kind of topology you choose depends on the number of computers you have to connect, how quickly you want communications to proceed, and how you want to distribute the task of passing along communications. A star topology provides fast communication between computers, but requires both a large portion of the central computer's total operation time and a large number of serial lines on the central computer. A serial topology distributes the communication burden evenly, requiring only two serial lines per computer, but it is slow if the chain is very long (communication between computers can take several minutes). Often a combination of star and serial topologies makes the best network. In any case, make the choice you think best. If you discover you have made a wrong choice, you may change the network at any time.

DRAWING A NETWORK TOPOLOGY MAP

A network topology map is a sketch of the connections between computers in the network. You use the map to plan the number and location of the serial lines you use to make the network.

You can make the map while you work out the topology. Simply arrange the machine names of each computer in the network on paper, then mark each pair of computers you wish to connect with serial lines. For example, the topology map for three computers might look like this:

```
a ---------- b ---------- c
```

As you draw, make sure that there is no more than one connection between any two computers in the network. Furthermore, make sure that no rings are formed (a ring is a series of connections that form a closed circle). Multiple connections and rings are not permitted.

ASSIGNING LINES AND SPEEDS.

Once you have made the topology map, you can decide which serial lines to use. Since every connection between computers in the network requires exactly two serial lines (one on each computer), you need to be very careful about assigning the lines. Follow these steps:

1. Make a list of the serial lines (TTY lines) available for use on each computer in the network. You can see a list of the serial lines on a computer by displaying the file /etc/ttys. A line is available if it is not connected to any device such as a terminal or modem.

2. Using the topology map, pick a computer, then assign one, and only one, serial line to each connection shown for that computer. The serial lines must be from the list of available lines for that computer. No line may be assigned more than once. For example, if computer a has only one available serial line (tty01), then the topology map should look like this:

```
      a ---------- b ---------- c
tty01
```

3. Repeat step 2 for all computers in the topology map. Make sure that each connection is assigned a line and that no two connections on any given computer have the same line. When finished, the map should look like this:

```
    a ---------- b ---------- c
tty01    tty02  tty03     tty04
```

   If a computer does not have enough available serial lines to meet its needs, you can make the lines available by removing the devices already connected to them. If you cannot remove devices, you must redraw your topology map.

4. Using the topology map, assign a serial line transmission speed for each computer pair. The speed may be any in the normal range for XENIX serial lines (typically 110 to 9600). Transmission speeds are a matter of preference. In general, a higher speed means a smaller amount of time to complete a transmission, but a greater demand on system's input and output capabilities. In some cases, transmission speeds are a matter of hardware capabilities. Some hardware is not

capable of transmission speeds greater than 1200 baud. For this reason, 1200 is the recommended speed when first installing Micnet. You may then increase the speed if you find the hardware can support it.

5. After the topology map is completely filled in, make a list of all computer pairs, showing their machine names, serial lines, and transmission speeds. You will use this list when installing the network.

CHOOSING ALIASES

Once you have decided how to connect the computers in the network, you can choose aliases for users in the network. An alias is a simple name that represents both a location (computer) and a user. Aliases are used by the mail command to allow you to refer to specific computers and users in a network without giving the explicit machine and user names. Although not a required part of the network, aliases can make the network easier to use and maintain.

There are three kinds of aliases: standard, machine, and forward. A standard alias is a name for a single user or a group of users. A machine alias is a name for a computer or an entire network (called a site). A forward alias is a temporary alias for a single user or group of users. A forward alias allows users who normally receive network communications at one computer to receive them at another.

When you build a network with the netutil program, you will be asked to provide standard aliases only. (You can incorporate machine and forward aliases into the network at your leisure.) Each standard alias must have a unique name and a list of the login names of the users it represents. You may choose any name you wish as long as it consists of letters and numbers, begins with a letter, and does not have the same spelling as the login names. The name should suggest the user or group of users it represents. The login names must be the valid login names of users in the network.

To help you prepare the aliases for entry during the netutil program, follow these steps:

1. Make a list of the user aliases (that is, the aliases that refer to just one user) and the corresponding login names of each user.

2. Make a separate list of the group aliases (that is, the aliases that refer to two or more users) and the login names or user aliases (from the first list) of the corresponding users. A group alias may have any number of corresponding users.

Note that there are a number of predefined group aliases. The name all is the predefined alias for all users in the network. The machine names of the computers in the network are predefined aliases for the users on each computer. Do not use these names when defining your own aliases.

BUILDING A NETWORK

You build a network with the netutil program. The program allows you to define the machines, users, and serial lines that make up the network.

To build a network, you must first create the Micnet files that define the network and then transfer these files to each computer in the network. After each computer receives the files, you may start the network and use it to communicate between computers.

The following sections describe how to build the network.

CREATING THE MICNET FILES

The Micnet files are created with the install option of the netutil program. The install option asks for the names, aliases, and serial lines of each computer in the network. As you supply the information, it automatically creates the files needed for each computer. These files can then be transferred to the other computers in the network with the save and restore options of netutil . This means you can build the entire network from just one computer.

To use the install option, follow these steps:

1. Log in as the super-user.

2. Type:

   ```
   netutil
   ```

   and press CR . The program displays the network utility menu. The install option is the first item in the menu.

3. Type the number 1 and press CR . The program displays the following message:

   ```
   Compiling new network topology
   Overwrite existing network files? (yes/no)?
   ```

   Type y (for "yes") and press CR to overwrite the files. You must overwrite the existing network files to create the new network. The first time you install the network, these files contain default information that need not be saved. If you install the system a second time or expand the system, it may be wise to save a copy of these files before starting the install option. The files can be saved on a diskette with the save option, described in the next subsection of this chapter.

   Once you have typed y , the program displays the following message:

   ```
   Enter the name of each machine
         (or press RETURN to continue installation)
    Machine name:
   ```

4. Enter a machine name by typing the name and press CR . You may enter more than one name on a line by separating each with a comma (,) or a space. After you have entered all the names, press CR to continue to the next step. The program displays the names you have entered and asks if you wish to make changes:

```
Machine names are: name1 name2 ... namen
Do you wish to make any changes? (yes/no)?
```

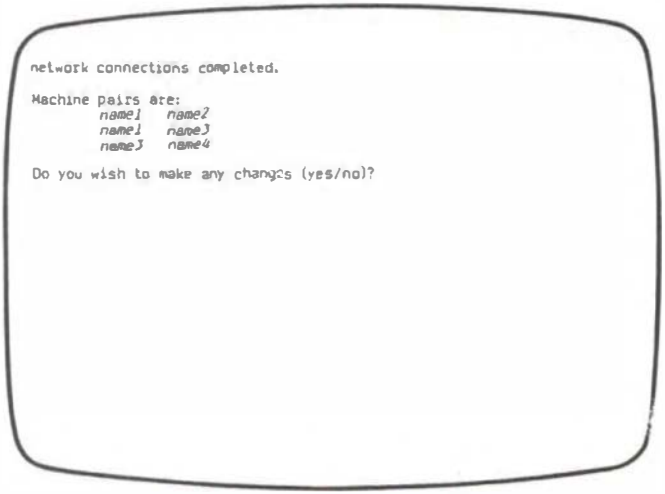
5. Type y (for "yes") if you wish to change the machine names. Otherwise, type n (for "no") or just press CR to move on to the next step. If you type n , the program displays the message:

```
For each machine, enter the names of the machines
        to be connected with it
Machine name:
  Connect to:
```

6. Using the list of machine pairs you created when planning the network, enter the machine names of the computers that connect to the computer specified by name. You may enter more than one name on a line by separating each name with a comma (,) or a space. When you have entered the machine names of all computers that connect to the specified computer, press CR .

7. The program prompts you to supply the machine names of the computers that connect to the next computer, and continues with this process until you have given the connections for all the machine names you entered in step 3. Follow the procedure described in step 6 for each machine name. As the program asks for each new set of connections, it will show a list of the machine names it already knows are connected with the current computer. You need not enter these names. The program automatically checks for loops. If it finds one, it ignores the machine name that creates the loop and asks for another.

   Finally, when you have given the connections for all computers in the network, the program displays a list of the connections and asks if you wish to make corrections:



8. Type y if you wish to enter the connections again. Otherwise, type n to move to the next step. If you type y , the program displays the message:

   For each machine pair, enter the tty name and tty speeds
   For the *name1* <==> *name2* machine pair.
   Tty on name1:

9. Using the list of serial line assignments you created when planning the network, type the serial line name or number (for example, tty03 or 3) for the first computer in the pair and press CR The program displays the message:

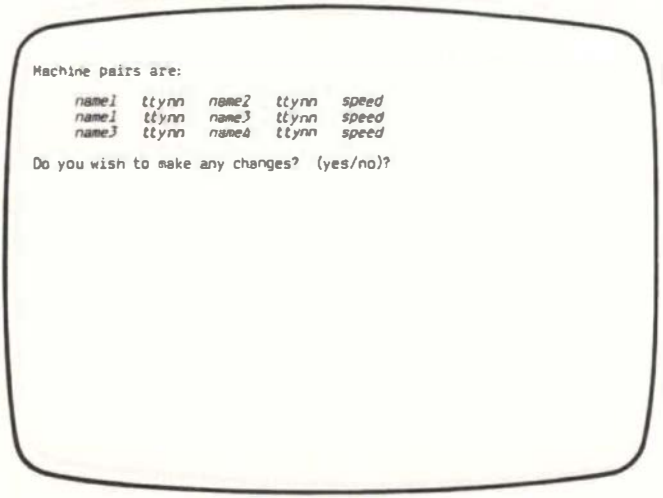
   Tty on *name2:*

10. Type the serial line name for the second computer in the pair and press CR . The program displays the message:

    Speed:

11. Type the speed (for example, 1200) and press CR .

12. The program prompts you to supply the serial lines and transmission speeds of the next pair, and continues with this process until you have given this information for all the remaining machine pairs. Repeat steps 8 through 11 for each machine pair. When you have given serial lines and speeds for all pairs, the program displays this information and asks if you wish to make corrections:



13. Type **y** if you wish to correct the serial lines and speeds by enteri g them again. Otherwise, type **n** to move to the next step. The program displays the message:

    Enter the names of users on each machine:

    For machine *name1:*
    Users on *name1:*

14. Enter a name by typing the login name of a user on the given computer, then press **CR** . You may enter more than one name on a line by separating each name with a comma (,) or a space. When you have entered all names for the given computer, press **CR** . The program displays the names of the users on the computer and asks if you wish to make corrections:

    Users on *name1* **are:**

    user1 user2 ... usern

    Do you wish to make any changes? (yes/no)?

15. Type **y** if you wish to correct the user names by entering them again. Otherwise, type **n** . If you type **n** , the program asks for the user names on the next computer.

16. Repeat step 14 and 15 for all remaining computers. When you have given names of users for every computer, the program asks if you wish to enter aliases:

    Do you wish to enter any aliases? (yes/no)?

17. Type y if you wish to enter aliases. Otherwise, type n to complete the installation. If you type y , the program displays the message:

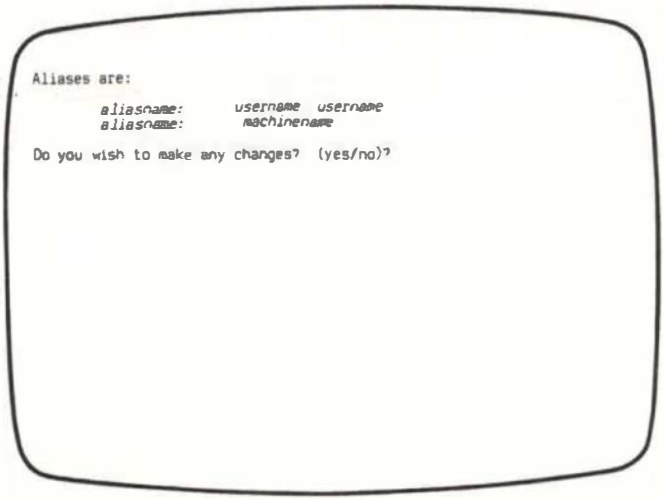
```
Each alias consists of two parts, the first is the alias
name, the second is a list of one or more of the following:
        valid user names
        previously defined aliases
        machine names

 Alias:
```

18. Using the list of aliases you created when you planned the network, type the name of an alias and press CR . The program displays the message:

    Users/Aliases:

19. If the alias names a single user, type the login name of that user and press CR . If, on the other hand, the alias names several users, type the login names of the users. If one or more of the users named by the alias are already named by other aliases, type the aliases instead of the login names. If all the users on one computer are named by the alias, type the machine name instead of the login names. In any case, make sure that each item typed on the line is separated from the next by a comma (,) or a space. If there are more items than can fit on the line, type a comma after the last item on that line and press CR /&. You can then continue on the next line. After all names and aliases have been typed, press CR . The program asks for another alias.

20. Repeat steps 18 and 19 for all remaining user aliases in your list. When you have given all aliases, press CR . The program displays a list of all aliases and their users and asks if you wish to make corrections:



21. Type **y** if you wish to enter all aliases again. Otherwise, type **n** to complete the installation.

Once you direct netutil to complete the installation, it copies the information you have supplied to the network files, displaying the name of each file as it is updated. Once the files are updated, you may use the save option to copy the Micnet files to a diskette.

SAVING THE MICNET FILES

You can save copies of the Micnet files on a diskette with the save option of the netutil program. Saving the files allows you to transfer them to the other computers in the network. Before you can save the files, you need to format a diskette (see Chapter 5, "Using File Systems").

To save the files, follow these steps:

1. Log in as the super-user.

2. Type:

   ```
   netutil
   ```

   Press **CR** . The program displays the network utility menu.

3. Type the number 2 and press **CR** . The program displays the message:

   ```
   Save to /dev/fd0 (yes/no)?
   ```

4. If you wish to use diskette drive 0, insert a blank, formatted diskette into the drive, wait for the drive to accept the diskette, then type y (for "yes") and press CR . If you do not wish to use drive 0, type n (for "no")and press CR . The program displays a prompt asking you for the filename of the diskette drive you wish to use. Insert a blank, formatted diskette into your chosen drive, wait for the drive to accept the diskette, then type the special filename of the drive.

   In either case, the program copies the Micnet files to the diskette.

5. Remove the diskette from the drive. Using a soft tip marker (do not use a ball point pen), label the diskette "Micnet diskette."

As soon as all files have been copied, you can transfer them to all computers in the network.

RESTORING MICNET FILES

The last step in building a Micnet network is to copy the Micnet files from the Micnet disk to all computers in the network. Do this with the restore option of the netutil program. For each computer in the network, follow these steps:

1. Log in as the super-user.

2. Type:

   ```
   netutil
   ```

   Press CR . The program displays the network utility menu.

3. Type the number 3 and press . The program displays the message:

   ```
   Restore from /dev/fd0 (yes/no)?
   ```

4. If you wish to use diskette drive 0, insert the Micnet diskette into the drive, wait for the drive to accept the diskette, then type y (for "yes") and press CR . If you do not wish to use the drive, type n (for "no") and press CR . The program displays a prompt asking you for the filename of the diskette drive you wish to use. Insert the Micnet diskette into your chosen drive, wait for the drive to accept the diskette, then type the special filename of the drive.

   In either case, the program copies the network files to the appropriate directories, displaying the name of each file as it is copied. Finally, the program displays the message:

   ```
   Enter the name of this machine:
   ```

5. Type the machine name of the computer you are using and press CR . The program copies this name to the new /etc/systemid file for the computer. If necessary, it also disables the serial lines used on

the computer, preparing them for use with the network.

When the files have been copied, you may start the network with the start option.

STARTING THE NETWORK

Once the Micnet files have been transferred to a computer, you can start the network with the start option of the netutil program. The start option starts the Micnet programs that are needed to communicate between the computers in the network.

To start the network, follow these steps for each computer in the network:

1. Log in as the super-user.

2. Type:

   ```
   netutil
   ```

   Press CR . The system displays the network utility menu.

3. Type the number 4 and press CR . The program searches for the /etc/systemid file. If it finds the file, it starts the network. If not, it asks you to enter the machine name of the computer and then creates the file. The program also asks if you wish to log errors and transmissions. In general, these are not required except when checking or testing the network. When starting the network for the first time, type n (for "no") in answer to each question and press CR .

Once the network has started, you may move to the next computer and start the network there.

Note that, for convenience, you can let each computer start the network automatically whenever the system itself is started. Simply include the command:

```
netutil start
```

in the system initialization file, /etc/rc, of each computer. To add this command, use a text editor as described in Chapter 10, "Solving System Problems". You can add the -x or -e options to this command line if you wish to log transmissions or errors. Even if you do not use these options, Micnet copies a log in and log out message to the system LOG file each time you start and stop the network. This means you will need to periodically clear the file. See Chapter 6, "Maintaining File Systems".

TESTING A MICNET NETWORK

After you have started a network for the first time, you should test the network to see that it is properly installed. In particular, you must determine whether or not each computer is connected to the network.

To test the network, you will need to know how to use the mail command (see the XENIX User Guide). The following sections explain how to test the network and how to correct the network if problems are discovered.

CHECKING THE NETWORK CONNECTIONS

You can make sure that all computers are connected to the network by mailing a short message to all (the alias for all users in the network) with the mail command. Follow these steps:

1. Choose a computer.

2. Log in as the super-user.

3. Use the mail command (see the XENIX User Guide) and the all alias to mail the message "micnet test" to all users in the network.

4. Check the mailboxes of each user in the network to see if the message was received. To check the mailboxes, log in as the super-user at each computer and use the cat command to display the contents of each user's mailbox. The name of each user's mailbox has the form:

   /usr/spool/mail/login-name

   where login-name is the user's login name.

If all users have received the message, the network is properly installed. If the users at one or more computers fail to receive the message, the computers are not properly connected to the network. To fix the problem, you need to locate the computer which has failed to make a connection. The next section explains how to do this.

USING THE LOG FILE TO LOCATE A PROBLEM

You can locate a connection problem by examining the LOG files on each computer in the network. The LOG files contain a record of the interaction between each pair of computers. There are two LOG files for each pair of computers (one file on each computer). The LOG files on any given computer are kept in subdirectories of the /usr/spool/micnet directory. Each subdirectory has as its name the machine name of the other computer in the pair. You can examine the contents of a LOG file by typing:

```
cat /usr/spool/micnet/remote/machine-name/LOG
```

Press CR . The machine-name must be the name of a computer that is paired with the computer you are using.

Each LOG file should contain a startup message which lists the name of each computer in the pair and the serial line through which the pair is connected. It also shows the date and time at which the network was started. The message should look like:

```
daemon.mn: running as MASTER
Local system: a
Remote system: b, /dev/tty02
Tue Sep 27 22:30:35 1983
```

A startup message is added to the file each time the network starts successfully. If the message is not present, this means that one or more of the network files and directories cannot be found. Make sure that you have used the restore option to transfer all the network files to the computer. Also, make sure that the /etc/systemid file contains the correct machine name for the given computer.

Each LOG file will contain a handshake message if the connection between the computer pair has been established. The message:

```
first handshake complete
```

is added to the file on a successful connection. If the message is not present, make sure that the network has been started on the other computer in the pair. The network must be started on both computers before any connection can be made. If the network is started on both computers but no handshake message appears, then the serial line may be damaged or improperly connected. Check the serial line to make sure that the cable is firmly seated and attached to the correct RS-232 connectors on both computers. If necessary, replace the cable with one known to work.

If both the startup and handshake messages appear in the LOG file but the network is still not working, then there is a problem in transmission. You can create a record of the transmissions and errors encountered while transmitting by restarting the network and requesting Micnet to log all transmissions and errors. Just type y (for "yes") when the start option asks if you wish to log errors or transmissions.

Error entries contain the error messages that were generated during transmission. Each message lists the cause of the error and the subroutine which detected the error. For example, the message:

```
rsync: bad Probe resp: 68
```

shows that the rsync subroutine received a bad response (character 68 hexadecimal) from the other computer. You may use this information to track down the cause of the problem. One common problem occurs when electronic noise passes stray information down the serial line. Make sure that the serial line's cable is properly protected against noise (for example, that the cable does not lie near any electric motor, generator, or other source of electromagnetic radiation). Also, make sure the cable is in good condition.

Transmission entries contain a record of normal transmissions between

computers. Each entry lists the direction, byte count, elasped time, and time of day of the transmission. For example, the entry:

```
rx: 0c 01 22:33:49
```

shows that 12 characters (0c hexadecimal) were received (rx) at 22:33:49. The elasped time for the transmission was 1 second. You can use the records to see if messages are actually being transmitted.

STOPPING THE NETWORK

You can stop the network with the stop option of the netutil program. This option stops the Micnet programs, which stops communication between computers in the network.

To stop the network, follow these steps on each computer in the network:

1. Log in as the super-user.
2. Type:

   netutil

   Press CR . The program displays the network utility menu.
3. Type the number 5 and press CR . The program stops the network programs running on the computer.

MODIFYING THE MICNET NETWORK

You can modify a Micnet network at any time by changing one or more of the Micnet files. You can reinstall the network with the netutil program. For very small changes (for example, correcting the spelling of an alias), you can modify the Micnet files directly with a text editor. The files and their contents are described in detail in Section (M) of the XENIX User and System Administrator Reference Manual.

Before making any changes to a file, you should make a copy. You can make a copy with the cp command. You can replace an old file with the updated file by using the mv command. Once one or more files have been changed on one computer, the files must be transferred to the other systems in the network using the save and restore options. These options can only be used after you have stopped the network.

Note that changes to the aliases file will not be incorporated into the system until the aliashash program is executed. This program produces the aliases.hash file the network needs to resolve aliases. See aliashash(M) in the XENIX User and System Administrator Reference Manual for a description of this command.

USING A UUCP SYSTEM

You can send and receive mail from other Micnet sites by installing a uucp system on one computer in your site. A uucp system is a set of XENIX programs that provide communication between computers by using ordinary telephone lines.

To use a uucp system with your Micnet network, follow these steps:

1. Choose a computer in the Micnet site.

2. Log in as super-user.

3. Install a uucp system on the computer. Installation of a uucp system requires a modem and the uucp software that is provided with the XENIX Software Development System. See the XENIX System and Application Software Development Tools User Guide for complete details.

4. Using a text editor, add the entry:

   uucp: *machine-name*

   to the maliases file of the computer on which the uucp system is installed. The *machine-name* is the name of the computer on which the uucp system is installed.

5. On each computer in your site, add the entry:

   uucp: *machine-name*

   to the maliases file. The *machine-name* must be the name of the computer on which the uucp system is installed.

You can test the uucp system by mailing a short letter to yourself via another site. For example, if you are on the site "chicago" and there is another Micnet site named "seattle" in the system, then the command:

```
mail seattle!chicago!johnd
```

will send mail to the "seattle" site, then back to your "chicago" site, and finally to the user "johnd" in your Micnet network. Note that a uucp system usually performs its communication tasks according to a fixed schedule, and may not return mail immediately.

12. BUILDING A COMMUNICATION SYSTEM

ABOUT THIS CHAPTER

This chapter explains how to build a communication system for your computer using a normal telephone line and a Hayes Smartmodem 1200.

CONTENTS

INTRODUCTION

This chapter explains how to build a communication system for your computer using a normal telephone line and a Hayes Smartmodem 1200. A communication system provides a way to:

- Log in to the computer from a remote terminal or computer.
- Use the cu command to call and log in to other computers.
- Use the uucp command to copy files to and from remote computers.
- Use the uux command to execute a remote mail program (rmail) on a remote computer.

In other words, the communication system is intended to give access to terminals and computers that cannot be connected to your computer through a direct serial line. In particular, the communication system is a practical solution to the problem of two Micnet networks that cannot be connected because of distance or cost of cable.

All communication tasks are supported by a variety of files and directories. In addition, the tasks invoked by the uucp and uux commands are actually performed by a system of underlying programs, called the uucp system. The files and underlying programs are described later in this chapter.

The following sections explain how to install the modem and prepare the programs you need to build a communication system. They also explain how to install and maintain a uucp system. Many of the administrative tasks associated with installing and setting up a uucp system are handled by the uuinstall utility. The next section explains how this utility is used. The last section of this chapter presents the program listing of the dial program used to communicate with the modem. This code may be used to create a modified dial program that communicates at a different baud rate (e.g., 300 baud) or with a different modem.

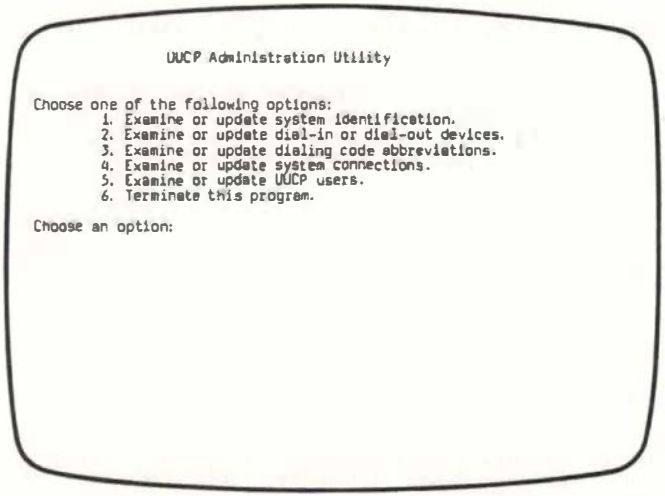
SETTING UP A UUCP SYSTEM

The uuinstall program is used to manage the information contained in the control files for the uucp system. There are five of these files:

/etc/systemid	Contains the name by which your system is known on the uucp network.
/usr/lib/uucp/L.sys	Contains a description of other systems you can call, or which call you, on the uucp network.
/usr/lib/uucp/USERFILE	Contains a description of the access you allow to other systems and users on those systems, and to the files of the local system.

/usr/lib/uucp/L-devices	Contains a description of the devices that are used to connect to other systems on the uucp network.
/usr/lib/uucp/L-dialcodes	Contains a list of abbreviations that are used in the dialing codes for placing calls to other systems.

To invoke the uuinstall utility, type ''uuinstall'' on the command line. The screen displays the main menu shown below:



The uuinstall utility will keep returning to this display after performing the action that you request. When you have completed all the required changes to the uucp control files, you should enter option 6. The uuinstall program will display:

Do you want to update the UUCP control files?

This gives you one last chance to decide whether you wish to make the changes to the control files that you have indicated. Typing "y" after this prompt will cause the control files to be updated. Any other response will cause the uuinstall program to terminate without making any changes to the files. Each of the options listed in the main uuinstall menu are described in detail later in this chapter.

You can also invoke the uuinstall program with a -r command line option. This allows you to read the current settings of the uuinstall menu options, but will not allow you to make any changes. In order to run uuinstall , you must be the super-user. Refer to uuinstall (C) in the XENIX User and System Administrator Reference Manual. for more information.

WHAT YOU NEED

To install a communication system on your computer, you will need:

- A Hayes Smartmodem 1200.
- A standard telephone jack for access to the telephone system (touch tone line required).
- An RS-232 serial line (or serial port) on your computer.
- An RS-232 cable to connect the serial line to the modem.

For proper operation of the modem, the RS-232 cable must provide the pin connections shown below. Note that the computer's serial connector must have a DTE (Data Terminal Equipment) configuration. The modem is assumed to have a DCE (Data Communications Equipment) configuration.

Pin Connections

Computer (DTE)	Modem (DCE)
1	1
2	2
3	3
6	6
7	7
8	8
20	20

These pin connections are explained in the *Hayes Smartmodem 1200 Reference Manual* (This manual will be referred to as the *Hayes Reference Manual* in this chapter.)

Make sure you inform the telephone company of your intent to use a modem with your telephone line. See the *Hayes Reference Manual* for details.

Finally, since many of the tasks you must perform require special permissions, you must log in to the super-user account before performing them.

INSTALLING THE MODEM

Installing the modem is the next step in creating a complete communication system. The installation has four steps:

1. Choose a serial line.
2. Set the dialing configuration.

3. Connect the modem.

4. Test the connection.

The following sections explain each step in detail.

CHOOSE A SERIAL LINE

You must choose the RS-232-serial line you wish to use with the system and connect it to the modem. If there are no lines available, you must install a new serial line or make one available by removing any device connected to it. If you remove a terminal, make sure no one is logged in.

Once you have chosen a serial line, find the name of the device special file associated with the line by looking in Appendix A of this guide. The filename should have the form:

/dev/tty *nn*

where *nn* is the number of the corresponding line. For example, */dev/tty00* usually corresponds to serial line 0. You need the filename for later steps.

SET THE DIALING CONFIGURATION

In this communication system, your modem can be used to both send and receive calls. You must set the appropriate switches on the modem. Follow these steps:

1. Remove the front cover of the modem and locate the 8-pin configuration switch. (See the *Hayes Reference Manual* for instructions on how to remove the cover and locate the switch.)

2. Set the pins on the configuration switch to the following positions:

pins	1	2	3	4	5	6	7	8
positions	up	up	down	down	up	up	up	down

3. Replace the front cover.

CONNECT THE MODEM

Once your modem's dialing configuration is set, you are ready to connect the modem to your computer. Review the installation instructions given in the *Hayes Reference Manual* , then follow these steps:

1. Connect the RS-232 serial cable to the serial line connector on the modem, then to the serial line connector on your computer. Make sure the cable is fully connected.

2. Plug the telephone line cable into the telephone connector on the modem, then into the telephone wall jack.

3. Plug in the power cord of the modem.

TEST THE MODEM

As the last step of the modem installation, you should test the modem to make sure that it can send and receive calls. Once you have verified that the modem is working, you can begin to use the communication system.

To test the modem, follow these steps:

1. Start the computer and log in as the super-user.

2. Disable the modem's serial line by typing:

   disable /dev/tty *nn*

   where *nn* is the serial line's number.

3. Turn on the modem's power.

4. Make sure the volume switch on the modem is at an appropriate level. You must be able to hear the modem to carry out this test successfully. See the *Hayes Reference Manual* for the location of this switch.

5. Invoke the **dial** program using a command line of the form:

   /usr/lib/uucp/dial /dev/tty *nn number speed*

   where /dev/tty *nn* is the filename of your serial line, and *number* is your telephone number (the number of the telephone jack your modem is connected to). For example, if your serial line is */dev/tty00* and your telephone number is 5551234, type:

   /usr/lib/uucp/dial /dev/tty00 5551234 1200

6. Listen carefully to the modem. You should hear each digit as the number is dialed; then you should hear the busy signal when the telephone system tries to make connection with your modem.

7. If you hear the busy signal, wait a few moments and listen carefully for the modem to hang up. The modem automatically discontinues any call it cannot connect.

8. If you don't hear the busy signal, make sure you have connected the modem to the telephone jack. Make sure the jack is connected to the phone system, and make sure you gave the correct number when invoking dial .

9. If you did not hear the modem dial, make sure the volume switch is up. See that the modem is connected to the correct serial line and that the cable connection is tight. Make sure you gave the correct filename when invoking dial , and verify that the modem's power is on.

CREATING A DIAL-IN LINE

You can create a dial-in line for remote terminals or computers by enabling the modem's serial line with the enable command after making sure your modem's serial line has an appropriate */etc/ttys* file entry. Once the line is enabled, any user at a remote terminal or computer can log in to your computer by calling your modem and following the ordinary login procedure. To create a dial-in line, follow these steps:

- Log in as the super-user.

- Use the cat command to examine the contents of the */etc/ttys* file.

- Locate the entry in this file that corresponds to your serial line. The correct entry contains the name of your serial line and must have the form:

  03tty *nn*

  where *nn* is the number of your serial line.

- If necessary, use a XENIX text editor to change the entry or create a new entry for your serial line. Make sure the first two digits in the entry are 0 and 3, respectively.

- Save the edited file and exit the editor.

- Type:

  enable /dev/tty *nn*

  where *nn* is the number of your serial line. This enables the line for logins.

Your computer will now receive calls from remote terminals or computers, and it will prompt for a login name.

CREATING A DIAL-OUT LINE

You can create a dial-out line by disabling the modem's serial line, creating the call unit files, and finally setting up the */usr/lib/uucp/L-devices* file. A dial-out line lets you call and log in to other computers by using the cu command. The cu command uses the */usr/lib/uucp/L-devices* file to locate the modem's serial line and set the proper line speed when these values are not given explicitly in the cu command line. The call unit files are actually specially named files that are linked to your modem's serial line and used by uucp system programs.

The following sections explain how to create the necessary files and enable your line.

Your modem's serial line cannot be both dial-in and dial-out at the same time. However, you can alternate between dial-in and dial-out at different times of the day by enabling or disabling the serial line as needed. Make sure you wait at least one minute between each invocation of the enable and disable commands.

CREATE THE CALL UNIT FILES

You must create two new device files, called ''call unit'' files, using the ln command and your serial line file. Follow these steps:

1. Log in as the super-user.

2. Check for any existing call unit files by using the l command. Type:

   ```
   l /dev/cu*
   ```

   and examine the output. Call unit filenames have the form:

   ```
   /dev/cua n
   ```

   and:

   ```
   /dev/cul n
   ```

   where *n* is the same number as the corresponding serial line. If these files exist, you can skip the next step and continue with setting up the */usr/lib/uucp/L-devices* file.

3. Use the ln command to create the call unit files. For example, if your serial line is named /dev/tty00, type the commands:

   ```
   ln /dev/tty00 /dev/cua0
   ```

   and:

   ```
   ln /dev/tty00 /dev/cul0
   ```

4. Use the chmod command to change the access mode of the call unit files to read and write for everyone. For example, the command:

```
chmod ugo+rw /dev/cua0 /dev/cul0
```

sets the appropriate permissions for the */dev/cua0* and */dev/cul0* files.

SET UP THE L-DEVICES FILE

The */usr/lib/uucp/L-devices* file defines the devices you intend to use to implement the dial-out line. The file is also used by programs in the uucp system (as described later). The file contains one or more entries of the form:

type line call-unit speed

where *type* must be ''ACU'' if you are using an automatic call unit (modem) or ''DIR'' if you are using a direct serial line, *line* and *call-unit* are the */dev/cul* and */dev/cua* call unit filenames, respectively, and *speed* is the line speed or baud rate for transmissions. The call unit files are assumed to be in the */dev* directory, so the full pathname is not required. For example, the following entry:

```
ACU  cul0  cua0  1200
```

defines the */dev/cul0* device as the line, */dev/cua0* as the call-unit, and 1200 as the line speed.

With the Hayes modem, the *speed* must be set to 1200. Note that if you adapt the communication system for other purposes and give a *line* to a hardwired device (e.g., a direct serial line to another computer), you must use the number 0 for the *call-unit* field instead of a device name.

Use the uuinstall program to set up this file. When you invoke the uuinstall program, select option 2 when the main menu is displayed. You will then be asked if you wish to see the current devices. Type ''y'' and a screen similar to the following one will appear:

```
#       Type    Line      Call-Unit       Speed

0.      ACU     cul0        cua0          1200
1.      DIR     tty00                     1200
```

This display corresponds to the two lines in the L-devices file which would be:

```
ACU cul0 cua0 1200
DIR tty00 0 1200
```

The program now asks whether you wish to add or delete an entry in the file. Entries are always deleted by specifying the entry number (#)

shown in the first column on the screen display. If you request that an entry be added, you will be prompted for the type of the unit, which can be an ACU or a direct line (respond "a" for ACU or "d" for a direct line). If you simply press CR in response to this prompt, the uuinstall display will return to the main menu.

If you specify an ACU, you will then be prompted for the unit number of the calling unit and the line. Respond with just the number in each case; the uuinstall program will generate the appropriate "cua" and "cul" strings. If you specify a direct line, you will be prompted for the line number. Finally, you will be prompted for the speed of the line. Your response is checked and, if it is invalid, the program prompts you for a valid response.

ENABLE THE SERIAL LINE

Enabling the serial line is the last step in creating a dial-out line. To enable the serial line, follow these steps:

1. Make sure your modem has been installed and tested.

2. Make sure you are logged in as the super-user.

3. Disable the modem's serial line by typing:

   disable /dev/tty *nn*

   where *nn* is the number of your modem's serial line. If the line is already disabled, the command displays an error message that you can safely ignore.

You will now be able to call other computers that have dial-in lines by using the cu command. For a complete description of the command, see cu (C) in the XENIX User and System Administrator Reference Manual.

INSTALLING A UUCP SYSTEM

A uucp system is a set of files and programs that let you use the uucp and uux commands to transfer files and commands between computers connected by your communication system. Before you can use the uucp and uux commands, you must install the uucp system by modifying a number of uucp system files using the uuinstall utility program.

The uucp system actually provides two different methods of interaction with other computers. One method requires a dial-in line through which remote computers can log in and transfer files and commands. With this method, your computer is called a "dial-in site". The other method requires a dial-out line through which your computer can call other computers. With this method, your computer is called a "dial-out site". Each method requires its own set of uucp system files.

Although you can install files for both methods, you can only use one method at a time. It is possible, however, to use alternate methods at

different times of the day by creating a shell script that automatically enables or disables the line to permit dialing in or dialing out.

The following sections explain how to set up the files for both methods of interaction. They also explain how to create a transmission schedule and develop a cron script to implement the schedule.

CHOOSE A UUCP SITE NAME

In a uucp system, every computer belongs to a given "site". A site is any computer or any Micnet network that can communicate with the uucp system through a modem. To distinguish one site from another, every site must have a unique "site name". A site name is any combination of letters and digits that begins with a letter and is no more than seven characters long. The site name may then be used in **uucp** and **uux** commands to direct transmissions to the appropriate computer or Micnet network.

The site name should suggest some characteristic of the site, such as its location or affiliation. For example, a site in Chicago can be named "chicago", or a site in the legal department can be named "legal". The site name must be unique. That is, no other computer that calls your computer or is called by your computer can have the same site name.

Once you have chosen a site name, you will need to add it to the */etc/systemid* file as described in the next section.

SET UP THE SYSTEMID FILE

Each site must have a */etc/systemid* file. The file defines the site name of the given site and associates the site with a Micnet network if any. The file has the form:

sitename
[*machinename*]

where *sitename* is the name of the given site, and *machinename* is the Micnet machine name for that computer. The machine name is optional only if the computer is not connected to a Micnet network. For example, the entries:

chicago
brewster

define a site named "chicago", whose Micnet machine name is "brewster".

Since uucp systems are often created after a Micnet network has been established, the *systemid* file usually exists already on a given site. In this case, you must add the site name to the beginning of each *systemid* file on each computer in the Micnet network. Note that you may give more than one machine name if desired, but each name must be on a separate line. For a full description of the *systemid* file, see **systemid (M)** in the XENIX User and System Administrator Reference Manual.

You can set up the *systemid* file by choosing option 1 of the uuinstall program. The current site name and m chine name will be displayed and you will be prompted to select a change to either of these names. If you do wish to change either or both of them, you will be prompted to enter the new names that you have chosen.

CREATE A DIAL-IN SITE

You can create a dial-in site by installing the uucp login information required by other computers that wish to log in and transfer files and commands. This information consists of the following:

- One or more */etc/passwd* file entries.
- User access information in the */usr/lib/uucp/USERFILE* file.
- One or more */usr/lib/uucp/L.cmds* file entries.

You can create this information by using the **mkuser** command to set up login information in the */etc/passwd* file, using the *uuinstall* command to set up the user access information, and using an editor to view and if necessary modify the contents of the */usr/lib/uucp/L.cmds* file. The following sections explain the required format of the information.

Once the information is installed, you can enable the system for logins by creating a dial-in line.

Create uucp login entries

A dial-in site must provide a login entry for the sites that call it. These entries must be placed in the */etc/passwd* file. A uucp login entry has the same form as an ordinary user login entry but gives a special login directory and login program instead of the normal user directory and shell. To create a uucp login entry, follow these steps:

1. Choose a new login name and a user ID for the uucp login. The name may be any combination of letters and digits that is no more than eight characters long. Make sure the name is unique; a uucp login entry must not have the same name as any other login entry.
2. Use the **mkuser** command to add the password file entry for this new login name.
3. When **mkuser** prompts you for the shell to be used by this user, you should choose */usr/lib/uucp/uucico* for the shell.

Note that you can create new login entries for each site that calls your site, or use one entry for all sites.

Set up the user file

The */usr/lib/uucp/USERFILE* file defines which directories a given site (or a given user) may access using the uucp and uux commands. You should create one *USERFILE* entry for each site or user with a login entry in the */etc/passwd* file. Each entry has the form:

login , sitename pathname ...

where *login* is the login name for a given site, *sitename* is the site name of a given site, and *pathname* is the full pathname of the directory the given site may access. More than one *pathname* may be given if desired. The *login* and *sitename* are optional.

The following rules explain how access is granted for each entry:

- A calling site is granted access to those directories defined in an entry containing its site name.

- A calling site whose name does not appear in an entry is granted access to the directories defined for the first entry with no site name.

- A user is granted access to those directories defined in an entry containing his or her login name.

- A user whose login name does not appear in an entry is granted access to directories defined in the first entry with no login name.

You may have more than one entry with the same login name if you wish, but you must make sure that at least one of these entries also has the site name of any calling site which can log in with that name, or that one of these entries has no site name.

For example, consider the following entries:

```
uuccg,chicago  /usr /usr2/market
uucp,    /usr/vendor
schmidt, /usr2/market /usr/vendor
,        /usr/spool/uucp/public
```

The site named *chicago* has access to files in the directories named */usr* and */usr2/market* . Any other site will be granted access to */usr/vendor* only. A local user named *schmidt* is granted access to the directories */usr2/market* and */usr/vendor* . All other users have access to */usr/spool/uucp/public* only.

This information can be set up by choosing option 5 of the uuinstall command. You will be prompted as to whether you need to see the current entries in the userfile. If you type "y", the screen display will be similar to the following:

```
#       Login           Sitename        Paths

0.      ANYLOGIN        ANYSITE /tmp
1.      ANYLOGIN        microso /tmp /usr/spool/uucp
```

The *ANYLOGIN* and *ANYSITE* entries are special entries displayed whenever a blank login name or site name field is encountered in the userfile.

You are then asked whether you wish to add or delete an entry in the file. Entries are always deleted by specifying the entry number (#) in the first column on the screen display. If you request that an entry be added, you will be prompted for the login name, site name, and pathname for the new entry. If you simply press CR in response to this prompt, the display will return to the main uuinstall menu.

In response to the requests for a login name and a site name, you may enter the special name "A" (meaning "ANY") which corresponds to a blank field in the userfile for the login or site names. The prompts for pathnames will continue until you enter a blank line.

Modify the commands file

The */usr/lib/uucp/L.cmds* file contains a list of the commands that can be invoked by means of **uucp** at a dial-in site.

The default content of the */usr/lib/uucp/L.cmds* file is as follows:

```
rmail
```

This means that only remote mail can be performed at the dial-in site.

You should use an editor to check the content of this file and if necessary modify it to include the commands you wish to have available. Entries should consist of command names, each appearing on a separate line. For example, the entries

```
rmail
cp
lpr
```

would allow mail, file copy and print operations from a remote site.

CREATE A DIAL-OUT SITE

You can create a dial-out site by installing the dialing information needed by your system to call and log in to other computers. This information consists of the following:

- Dialing abbreviations for remote computers in the */usr/lib/uucp/L-dialcodes* file.

- Information about logins on remote computers in the */usr/lib/uucp/L.sys* file.

- A transmission schedule in the form of a shell script to be called periodically by the cron program.

The following sections explain the required format of the information.

Once the information is installed, you can enable the system for calling other computers by creating a dial-out line.

Set up the L-dialcodes file

The */usr/lib/uucp/L-dialcodes* file defines abbreviations for often-used telephone prefixes and area codes. You may use these abbreviations in the *L.sys* file when forming the telephone numbers of remote sites.

The *L-dialcodes* file may contain one or more entries of the form:

abbreviation dial-sequence

where *abbreviation* is any combination of letters and digits that begins with a letter, and *dial-sequence* is any combination of digits that represents a telephone prefix, area code, or any other part of a telephone number. For example, the entry:

```
ms 555
```

defines the abbreviation "ms" to be the telephone prefix "555".

This information can be set up by choosing option 3 of the **uuinstall** command. You will be prompted as to whether you need to see the currant entries in the dialcodes file. If you type "y", the screen display will be similar to the following:

```
#       Abbreviation    Code

0.      Sea             206
1.      London          011441
```

You are then prompted to add or delete an entry in the file. Remember that you delete entries by specifying the entry number (#) in the first column on the screen display. If you request that an entry be added, you will be prompted for the abbreviation and the dialing code for each entry. If you press CR in response to this prompt, the display will return to the uuinstall main menu.

Set up the L-sys file

The */usr/lib/uucp/L.sys* file defines the names, telephone numbers, and login information of all sites in the system. The file contains one or more entries of the form:

sitename time device speed phone login

where *sitename* is the name of the site to be called; *time* is a combination of letters and digits that gives the weekdays and times when the given site can be called; *device* is the name of the device through which the given site is to be called; *speed* is the line speed for the call, *phone* is the phone number of the given site, and *login* is the login information required to log in to the given site. With the Hayes modem, the *speed* must be 1200.

The *time* defines when the given site can make calls to other sites. It has the form:

days times

where *days* is a list of one or more days of the week, and *times* is a range of times of day. The days of the week may be ''Su'', ''Mo'', ''Tu'', ''We'', ''Th'', ''Fr'', ''Sa'', ''Wk'' (for any weekday), ''Any'' (for any day), and ''Never'' (for call by special request only). The time of day must be given as a four digit number. The first pair of digits gives the hour (in terms of a 24 hour clock), the second pair gives the minutes. A range of times is a pair of times of the day separated by a hyphen (-). For example, the entry:

MoTuTh0800-1230

allows the given site to be called any Monday, Tuesday, or Thursday from 8 in the morning to 12:30 in the afternoon.

The *device* must be the keyword ''ACU'' if you are using a modem. If you are using a direct line to the other site, then you must give the filename of the serial line (or other device) you intend to use (e.g., tty01).

The *phone* must be the telephone number of the given site. It must have the correct number of digits (including area code if necessary) or be a combination of *L-dialcodes* abbreviations and digits. *L-dialcodes* abbreviations must go before any digits. Hyphens must not be used. For example, ''5551234'' is a valid local number and ''2065551234'' is a valid long distance number. If the abbreviation ''ms'' is defined to be ''555'', then ''ms1234'' may be used in place of ''5551234''.

With the Hayes modem, you may use a comma (,) in a number to cause a delay when dialing. This is useful if you must dial for an outside line before placing the call. For example, the number ''9,5551234'' causes a delay immediately after the ''9'' has been dialed. After the delay, the rest of the number is dialed. If you are not using a modem, then *phone* must be the filename of the device you intend to use instead of a phone number.

The *login* must be a sequence of names, numbers, and other information that represents the steps required to log in to the given site. This sequence has the form:

expect send [*expect send*] ...

where *expect* is the prompt or message that you expect the given site to return to the calling site, and *send* is the name, number, or other information that you wish to send in response to the expected prompt or message. For example, the following is the login sequence for a typical XENIX site:

```
login: uuccg ssword: market
```

Note that ''ssword:'' is given instead of the complete prompt ''Password:'' . Only the last eight characters in each expected prompt or message are examined, so you do not need to give the preceding characters if you wish to save space.

If you anticipate problems during the login sequence, you may include a conditional response immediately after each expected prompt or message. This conditional response has the form:

expect [- *send* - *expect1*] ...

where *expect* is the prompt or message you expect the given site to return; *send* is the name or number you wish to send if the prompt or message returned is not correct, and *expect1* is the prompt or message you expect after sending the conditional response. For example, the following command line shows you how to invoke the ''login'' prompt if it is not immediately present:

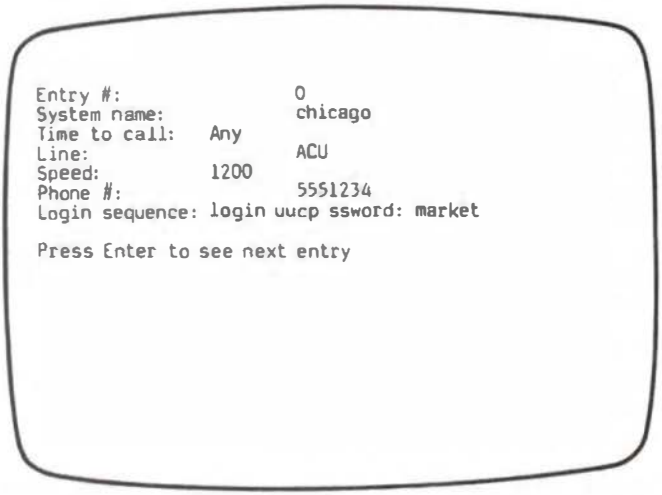
```
--login-EOT-login-uuccg ssword: market
```

There are two special keywords that you may use in the login sequence. The **EOT** keyword causes an end of transmission character to be sent, and the **BREAK** keyword causes a break character to be sent. (The break character is simulated using line speed changes and null characters and may not work on all devices and/or systems.)

The complete *L.sys* entry must be placed on one line as shown by the following example:

```
chicago Any ACU 1200 5551234 login uucp ssword: market
```

You can set up this information by choosing option 4 of the **uuinstall** command. (Refer to Section B.2.) You will be prompted as to whether you need to see the current entries in the *L.sys* file. If you type ''y'', the screen display will be similar to the following:



A new entry will be displayed each time you press the CR key. You are then asked whether you wish to add or delete an entry in the file. Once again, you can delete entries by specifying the entry number (#), which is in the first field for each displayed entry. If you request that an entry be added, you will be prompted for each field in turn. If you press CR in response to this prompt, the display will return to the uuinstall main menu.

The response to the prompt concerning the line to use for the call, can be either ''A'', which denotes an ACU, or the device number for the *tty* device to be used for the connection.

Create a transmission schedule

In the uucp system, the uucico program carries out all transmissions between your site and other sites, sending and receiving files and commands as long as there is work for it to do. On a dial-in site, uucico is always started whenever a calling site logs in, but on a dial-out site, uucico is only started when an explicit invocation of the program is given. This means you must invoke the program periodically on a dial-out site to ensure that all transmissions requested by the uucp and uux programs are completed. You can do this in one of two ways: invoke the program manually whenever you need it, or create a shell script and let the cron program invoke uucico automatically according to a schedule of transmissions.

The most convenient method is to let cron invoke uucico for you. To do this, you choose a schedule of times for uucico to be invoked, then create a */etc/crontab* file entry for this schedule. A */etc/crontab* entry has the form:

minutes hour day month day-of-week command-line

where *minutes* , *hour* , *day* , *month* , and *day-of-week* give the exact day of the year and time of day to execute the given *command-line* . Each item, except the *command-line* , must be an integer number within an acceptable range, (e.g., 0 to 59 for *minutes*) . A sequence of values for one item may be given by separating the values with commas. Also, you may use an asterisk (*) to represent all acceptable values. The *command-line* must be the name of the shell script you have created to invoke uucico .

You can add an entry to the */etc/crontab* file by using a XENIX text editor. For more information about the file, see cron (C) and crontab (C) in the . For example, the entry:

```
15,45 * * * * /usr/lib/uucp/transmit
```

invokes the shell script ''transmit'' every 30 minutes to sites for which requests are pending. The entry:

```
0 0 * * * /usr/lib/uucp/transmit
```

invokes ''transmit'' every day at midnight, and the entry:

```
15 2,4,6 * * *  /usr/lib/uucp/transmit
```

invokes the script every day at ''2:15'', ''4:15'', and ''6:15'' in the morning.

A shell script is simply a text file that contains one or more XENIX commands. If your uucp system is acting as both a dial-in and dial-out site at different times, then the script should have the form:

```
#
[ /bin/disable /dev/tty nn ]
if ($status = = 0) then
sleep 20
/usr/lib/uucp/uucico -rl [ -s | -S ] sitename
[ /bin/enable /dev/tty nn ]
endif
```

where *nn* is the number of your modem's serial line, *sitename* is the name of the site you wish to call. Use the -s option if you wish to force a call to the given site even if no requests for transmissions exist on the calling site. Note that you can use the -S option in place of the -s option if you wish to ignore the range of calling times given in the *L.sys* file. Give one uucico command for each site you wish to call. If you want to call only those sites for which requests exist, give a single uucico command, but do not give the -s or -S option with the command. If your computer is strictly a dial-out site, then the enable and disable commands are not required.

For example, the script:

```
#
/bin/disable /dev/tty00
if ($status = = 0) then
        sleep 20
        /usr/lib/uucp/uucico -rl -s chicago
        /bin/enable /dev/tty00
endif
```

places a call to the ''chicago'' site after disabling the serial line. The line must be disabled in order to dial out on that line. Similarly, it must be enabled to allow subsequent calls from other computers.

You can create a shell script by using a XENIX text editor. For convenience, the script should be placed in the */usr/lib/uucp* directory and must be given execute permissions for everyone. Note that you can also add uucp maintenance programs to the script.

LINK MICNET SITES

To use a uucp system with your Micnet network, follow these steps:

1. Add the entry:

   uucp:

   to the *maliases* file of the computer on which the uucp system is installed.

2. For all other computers in your site, add the entry:

   uucp: *machinename:*

   to the *maliases* file. The *machinename* must be the name of the computer on which the uucp system is installed. This longer form of entry may also be used on the computer on which the uucp system is installed.

You can test the uucp system by mailing a short letter to yourself via another site. For example, if you are on the site ''chicago'', and there is another Micnet site named ''seattle'' in the system, then the command:

```
mail seattle!chicago!johnd
```

will send mail to the ''seattle'' site, then back to your ''chicago'' site, and finally to the user ''johnd'' in your Micnet network. Note that a uucp system usually performs its communication tasks according to a fixed schedule, and may not return mail immediately.

MAINTAINING THE SYSTEM

This section explains how to maintain the uucp system. In particular, it explains how to display and merge the content of uucp log files, how to remove old requests and files from the spool directories, and how to solve some common problems. It also includes some sample shell files, which are toward the end of the section.

You can automate some maintenance tasks by creating shell command files and initiating these files with *crontab* entries. Other tasks require manual modification.

DISPLAY AND MERGE LOG FILES

You can display a record of the transmissions requested and completed to a given site or user by using the uulog command. The command displays the contents of the individual log files created for a given site or user and merges these entries with the system log file, *LOGFILE* . The log files contain information about queued requests, calls to remote sites, execution of uux commands, and file copy results. The command has the form:

uulog -s *sitename* -u *user*

where *sitename* names the site whose log files are to be displayed, and *user* names the user whose log files are to be displayed. If you do not give a *sitename* , *user* log files for all sites and users are displayed. The command places the new log files at the beginning of the existing *LOGFILE* .

The log files are originally created in the */usr/spool/uucp* directory as individual files, but should be copied to the *LOGFILE* on a regular basis since they are not copied automatically. For example, the command:

uulog

merges all log files and displays their contents. The command:

uulog -schicago

merges only log files created for the site "chicago".

Note that the system *LOGFILE* should be removed periodically since it is copied each time new log files are put into the file.

CLEAN THE UUCP SPOOL DIRECTORY

You can remove unwanted uucp system files from the uucp spool directory by using the uuclean command. The command removes temporary data, *LOG* , system status, and lock files from the spool directory if they are more than a given number of hours old. The command has the form:

uuclean -d *dir* -m -n *hours* -p *pre* -x *n*

where -d *dir* names the directory to be scanned; -m causes mail to be sent to the owner of each file removed; -n *hours* gives the age in hours of files to be removed, -p *pre* causes files with the given prefix to be examined and removed, and -x *n* directs the command to give the *nth* level of debugging output. You may specify up to 10 file prefixes with the -p option. If -m is given, most mail will be sent to the owner of the uucp programs, since most files put into the spool directory will be owned by the owner of the uucp programs. This is a result of the setuid bit being set on these programs. The default number of *hours* is 72 (3 days).

The uuclean program should be run once a day. You can invoke it automatically by using a system daemon such as cron . The command:

uuclean -pTM

removes all temporary data files that are at least three days old. The command:

uuclean -pLCK -hl -m

removes all lock files that are at least an hour old and mails a list of each file removed to the owner.

The uuclean command may also be run as needed to remove unwanted files after a system crash or an aborted uucp program.

RECLAIM LOG FILES AFTER A CRASH

You can reclaim individual log files after a system crash by changing their access mode with the chmod command, and then using the uulog command. After a transmission failure or system crash, the individual log file for the transmission may be left with access mode 0222, making it impossible for the uulog command to read the file. To reclaim the log file, use chmod to change the access mode to 0666. You can then let uulog merge them with the *LOGFILE* .

RECLAIM DATA FILES AFTER A CRASH

You can check the status of files transmitted from a remote site and possibly reclaim some or all of the data lost during an aborted transmission by examining system data files. The data files contain the contents of files copied from remote sites. These files are kept temporarily in the */usr/spool/uucp* directory and their names have the form:

TM. *pid.ddd*

where *pid* is a process-id, and *ddd* is a sequential three digit number starting at zero for each invocation of uucico and incremented for each file received.

The temporary data files are normally moved to the requested destination immediately after the transmission finishes. However, if a transmission fails or the system crashes, the file remains in the spool directory. You can examine the contents of this file with the cat command. If desired, you can reclaim the file by moving it to a new location with the mv command. Leftover data files that cannot be reclaimed should be removed using the uuclean command.

CHECK THE TRANSMISSION STATUS

You can check the status of transmissions between sites in the uucp system by examining the system status files. System status files contain information about login, dialup, or sequence check failure, as well as the talking status when two machines are conversing. The files are kept in the */usr/spool/uucp* directory and their names have the form:

STST. *sitename*

where *sitename* is the name of the remote site.

Normally, system status files are removed after each successful transmission, but when a failure occurs, the uucp system copies information about the failure to the file and leaves it in the directory. This prevents the uucp system from making further calls to the given site for about an hour, or for sequence check failures, until the file is removed.

To examine the status, use the cat command to display the contents of the file. If problems with transmissions are detected, there may be a problem with the modem or with the serial line connected to the modem.

If a system status file has been left due to a program or system crash, the file may prevent all subsequent transmissions to the given site. In this case, you must remove the file before attempting further calls.

CHECK FOR LOCKED SITES OR DEVICES

You can make sure the uucp system is not intentionally preventing transmissions to a given site or through a given device by examining the system lock files. The uucp system creates a lock file for each site being called and for each device being used to call a site. Lock files prevent the uucp system from attempting to duplicate conversations with a given site or from placing multiple calls on the same device. The lock files are kept in the */usr/spool/uucp* directory and their names have the form:

LCK.. *str*

where *str* is either a site name or the name of the calling device.

Since lock files prevent all calls to a given site or through a given device; it is wise to make sure no unnecessary lock files are left in the directory. If a transmission has been aborted or the system has crashed, the lock files will prevent subsequent transmissions for about about 24 hours. If you wish to place a call before this time, you must remove the file using the uuclean command.

CREATE MAINTENANCE SHELL FILES

The uulog and uuclean command can be invoked automatically by placing the commands in a shell file and creating a *crontab* entry for the shell file. The system daemon, cron , will then invoke the commands at the given times and most of the simple maintenance will be performed. For example, you can create a shell file that removes *TM* , *ST* , and *LCK* files daily. You can also create a shell file to maintain *C* . or *D* . files for work which can not be accomplished for such reasons as bad phone number and login changes. In this case, the shell file should contain the following commands:

```
/usr/lib/uucp/uuclean -pTM -pC. -pD.
/usr/lib/uucp/uuclean -pST -pLCK -n12
```

Note that the -n12 option causes any *ST* and *LCK* files that are older than 12 hours to be deleted. An appropriate *crontab* entry must be created in order to invoke the shell file automatically.

DETAILS OF OPERATION

This section describes the details of uucp system program operation. It explains the processes used to create system communication and defines the files used to support the system.

UUCP PROGRAMS

The uucp system consists of four primary and two secondary programs. The primary programs are:

uucp — This program creates work and gathers data files in the spool directory for the transmission of files.

uux — This program creates work and execute files, and gathers data files for the remote execution of XENIX commands.

uucico — This program executes the work files for data transmission.

uuxqt — This program executes XENIX commands found in execution files.

The secondary programs are:

uulog — This program updates the log file with new entries and reports on the status of uucp requests.

uuclean — This program removes old files from the spool directory.

dial — This program directs the modem to dial a remote site.

UUCP DIRECTORIES AND FILES

During execution of the uucp programs, the uucp system uses files from the following three directories:

/usr/lib/uucp — This is the directory used for uucp system files and all executable programs other than uucp and uux .

/usr/spool/uucp — This is the spool directory used during uucp execution.

/usr/spool/uucp/.XQTDIR — This directory is used during execution of execute files.

Files are created in a spool directory for processing by the uucp daemons. There are three types of files used for the execution of work:

Data files — Contain data for transfer to remote sites.

Work files — Contain directions for file transfers between sites.

Execution files — Contain directions for XENIX command executions which involve the resources of one or more sites.

UUCP - SITE TO SITE FILE COPY

The uucp program is the user's primary interface with the system. The uucp program was designed to look like the cp command. The syntax is:

uucp [*option*] ... *source* ... *destination*

where *source* and *destination* may contain the *sitename!* prefix which indicates the site on which the file or files reside or where they will be copied.

The options interpreted by uucp are:

-d Make directories when necessary for copying the file.

-c Don't copy source files to the spool directory, but use the specified source when the actual transfer takes place.

-m Send mail on completion of the work.

The following options are used primarily for debugging:

-s *dir* Use the *dir* directory for the spool directory.

-x *num* Use *num* as the level of debugging output.

The destination may be a directory name, in which case the filename is taken from the last part of the source's name. The source name may contain special shell characters such as ''?*[]''. If a source argument has a *sitename!* prefix for a remote site, the filename expansion will be done on the remote site. For example, the command:

uucp *.c chicago!/usr/dan

sets up the transfer of all files whose names end with *.c* to the */usr/dan* directory on the "chicago" machine.

The source and/or destination names may also contain a *~user* prefix. This prefix translates to the login directory on the specified site. For names with partial pathnames, the current directory is prepended to the filename. Filenames with ''../'' are not permitted.

The command:

uucp chicago!~dan/*.h ~dan

sets up the transfer of files whose names end with *.h* in *dan 's* login directory on site, *usg* , to *dan 's* local login directory.

For each source file, the program will check the source and destination filenames and the site-designator of each to classify the work into one of five types:

1. Copy source to destination on local site.

2. Receive files from other sites.

3. Send files to remote sites.

4. Send files from remote sites to another remote site.

5. Receive files from remote sites when the source contains special shell characters as mentioned above.

After the work has been set up in the spool directory, the uucico program must be started to try to contact the other machine to execute the work.

Copy files to a local destination

A cp command is used to do type 1 work. The -d and the -m options are not honored in this case.

Receive files from other sites

For type 2 work, a one line work file is created for each file requested, and is put in the spool directory with the following fields, each separated by a blank:

[1] R

[2] The full pathname of the source or a ~user/pathname. The *~user* part will be expanded on the remote site.

[3] The full pathname of the destination file. If the *~user* notation is used, it will be immediately expanded to be the login directory for the user.

[4] The user's login name.

[5] A ''-'' followed by an option list. (Only the -m and -d options will appear in this list.)

Send files to remote sites

For type 3 work, a work file is created for each source file and the source file is copied into a data file in the spool directory. (A -c option on the uucp program will prevent the data file from being made. In this case, the file will be transmitted from the indicated source.) Pathnames are checked using the *USERFILE* to verify access to the requested directory. The fields of each entry are as follows:

[1] S

[2] The full pathname of the source file.

[3] The full pathname of the destination or ~user/filename.

[4] The user's login name.

[5] A ''-'' followed by an option list.

[6] The name of the data file in the spool directory.

[7] The file mode bits of the source file in octal print format (e.g., 0666).

Copy files between sites For type 4 and 5 work, uucp generates a uucp command line and sends it to the remote machine; the remote uucico executes the command line.

UUX - SITE TO SITE EXECUTION

The uux command is used to set up the execution of a XENIX command where the execution machine and/or some of the files are remote. The syntax of the uux command is:

uux [-] [*option*]... *command-string*

where *command-string* is made up of one or more arguments. All special shell characters such as ''<>|^'' must be quoted either by quoting the entire command string, or by quoting the character as a separate argument. Within the command string, the commands and filenames may contain a *sitename!* prefix. Any arguments which do not contain a ' will not be treated as files. (They will not be copied to the execution machine.) The - option is used to indicate that the standard input for the given command should be inherited from the standard input of the uux command. The only option is essentially for debugging: -x *num* directs the command to use *num* as the level of debugging output.

The command:

pr abc | uux - chicago!rmail joe

will set up the output of ''pr abc'' as standard input to a mail command to be executed on site *usg* .

Uux generates an execute file which contains the names of the files required for execution (including standard input), the user's login name, the destination of the standard output, and the command to be executed. This file is either put in the spool directory for local execution or sent to the remote site using a generated send command (type 3 above).

For required files which are not on the execution machine, uux will generate receive command files (type 2 above). These command files will be put on the execution machine and executed by the uucico program. (This will work only if the local site has permission to put files in the remote spool directory as controlled by the remote *USERFILE* .)

The execute file will be processed by the uuxqt program on the execution machine. It is made up of several lines, each of which contains an

identification character and one or more arguments. The order of the lines in the file is not relevant and some of the lines may not be present. Each line is described in the following paragraphs.

User Line

U *user site*

where the *user* and *site* are the requestor's login name and site.

Required File Line

F *filename real-name*

where the *filename* is the generated name of a file for the execute machine and *real-name* is the last part of the actual filename (contains no path information). Zero or more of these lines may be present in the execute file. The uuxqt program will check for the existence of all required files before the command is executed.

Standard Input Line

I *filename*

The standard input is either specified by a < in the command string or inherited from the standard input of the uux command if the - option is used. If a standard input is not specified, */dev/null* is used.

Standard Output Line

O *filename sitename*

The standard output is specified by a > within the command string. If a standard output is not specified, */dev/null* is used. (Note that the use of >> is not in effect.)

Command Line

C *command* [*arguments*] ...

The *arguments* are those specified in the command string. The standard input and standard output will not appear on this line. All required files will be moved to the execution directory (a subdirectory of the spool directory) and the XENIX command is executed using the shell. In addition, a shell PATH statement is prepended to the command line as specified in the uuxqt program.

After execution, the standard output is copied or set up to be sent to the proper place.

UUCICO - COPY IN, COPY OUT

The uucico program will perform the following major functions:

- Scans the spool directory for work.
- Places a call to a remote site.
- Negotiates a line protocol to be used.
- Executes all requests from both sites.
- Logs work requests and work completions.

uucico may be started by a system daemon, by the user (this is usually for testing), or by a remote site. (The uucico program should be specified as the shell field in the */etc/passwd* file for uucp logins.)

When started with the -rl option, the program is considered to be in MASTER mode. In this mode, a connection will be made to a remote site. If started by a remote site, the program is considered to be in SLAVE mode.

The MASTER mode will operate in one of two ways. If no site name is specified (the -s option not specified), the program will scan the spool directory for sites to call. If a site name is specified, that site will be called, and work will only be done for that site.

The uucico program must generally be started directly by the user or by another program, such as a shell script invoked by cron . There are several options used for execution:

-rl — Starts the program in MASTER mode. This is used when uucico is started by a program or cron shell.

-s*sitename* — Works only for site *sitename* . If -s is specified, a call to the specified site will be made even if there is no work for the site in the spool directory, but will only call when times in the *L.sys* file permit it. This is useful for polling sites which do not have the hardware to initiate a connection.

-S*sitename* — Works only for site *sitename* . If -S is specified, a call to the specified site will be made even if there is no work for the site in the spool directory. Unlike -s , this option ignores the call times for the *sitename* given in the *L.sys* file.

The following options are used primarily for debugging:

-d*dir* — Uses directory *dir* for the spool directory.

-x*num* — Uses *num* as the level of debugging output.

The next part of this section describes the major steps within the uucico program.

Scan for work

The names of the work-related files in the spool directory have the following format:

type . sitename grade number

where *type* may be ''C'' for copy command file, ''D'' for data file, ''X'' for execute file; *sitename* is the remote site; *grade* is a character, and *number* is a four digit, padded sequence number. For example, the file:

C.res45n0031

is a work file for a file transfer between the local machine and the ''res45'' machine.

The scan for work is done by looking through the spool directory for work files (files with prefix ''C.''). A list is made of all sites to be called. Uucico calls the site specified by the -s or -S option and processes the corresponding work files.

Call a remote site

The call is made using information from several files which reside in the uucp program directory. At the start of the call process, a lock is set to forbid multiple conversations between the same two sites. The lock filename has the form:

LCK.. *str*

where *str* is the device name. The file is in the */usr/spool/uucp* directory.

The site name is in the *L.sys* file. The information contained for each site is:

[1] Site name.

[2] Times to call the site (days-of-week and times-of-day).

[3] Device or device type to be used for call.

[4] Line speed.

[5] Phone number if field [3] is ''ACU'', or the device name (same as field [3]), if not.

[6] Login information (multiple fields).

The time field is checked against the present time to see if the call should be made.

The *phone number* may contain abbreviations (e.g., mh, py, boston) which get translated into dial sequences using the *L-dialcodes* file.

The *L-devices* file is scanned using device type and line speed fields from the *L.sys* file to find an available device for the call. The program will try all devices which satisfy these fields until the call is made or until no more devices can be tried. If a device is successfully opened, a lock file is created so that another copy of uucico will not try to use it. If the call is complete, the login information in the last field of *L.sys* is used to log in.

The conversation between the two uucico programs begins with a handshake started by the SLAVE site. The SLAVE program sends a message to let the MASTER program know it is ready to receive the site identification and conversation sequence number. The response from the MASTER is verified by the SLAVE, and if acceptable, protocol selection begins. The SLAVE program can also reply with a call-back required message in which case, the current conversation is terminated.

Select a line protocol

The remote site sends a message:

P *proto-list*

where *proto-list* is a string of characters, each representing a line protocol.

The calling program checks the protocol list for a letter corresponding to an available line protocol and returns a use protocol message. The message has the form:

U *code*

where *code* is either a one character protocol letter or ''N'' which means there is no common protocol.

Process work

The initial role of MASTER or SLAVE to the processing work is the mode in which each program starts. (The MASTER has been specified by the -r1 option.) The MASTER program does a work search similar to the one used in the section, ''SCAN FOR WORK'', above.

There are five messages used during the process work, each specified by the first character of the message. They are:

S Send a file.

R Receive a file.

C Copy complete.

X Execute a uucp command.

H Hangup.

The MASTER program will send ''R'', ''S'', or ''X'' messages until all work from the spool directory is complete, at which point an ''H'' message is sent. The SLAVE program will reply with the first letter of the request and either the letter ''Y'' or ''N'' for yes or no. For example, the message ''SY'' indicates that it is okay to send a fila.

The send and receive replies are based on permission to access the requested file/directory using the *USERFILE* and read/write permissions of the file/directory. After each file is copied into the spool directory of the receiving site, a copy-complete message is sent by the receiver of the file. The message ''CY'' will be sent if the file has successfully been moved from the temporary spool file to the actual destination. Otherwise, a ''CN'' message is sent. (In the case of ''CN'', the transferred file will be in the spool directory with a name beginning with ''TM''). The requests and results are logged on both sites.

The hangup response is determined by the SLAVE program using a work scan of the spool directory. If work for the remote site exists in the SLAVE's spool directory, an ''HN'' message is sent and the programs switch roles. If no work exists, an ''HY'' response is sent.

Terminate a conversation

When an ''HY'' message is received by the MASTER program it is echoed back to the SLAVE program and the protocols are turned off. Each program sends a final ''OO'' message to the other. The original SLAVE program will clean up and terminate. The MASTER program will proceed to call other sites and process work as long as possible or terminate if a -s option was specified.

UUXQT - UUCP COMMAND EXECUTION

The uuxqt program is used to process execute files generated by uux . The uuxqt program is started by the uucico program. The program scans the spool directory for execute files (prefix *X.*) . Each one is checked to see if all the required files are available, and, if so, the command line or send line is executed by the following shell command:

```
sh -c
```

which you use with the command line after you have opaned appropriate standard input and standard output. If a standard output is specified, the program will create a send command or copy the output file as appropriate.

SECURITY

In unrestricted uucp systems, once users log in to another site through the uucp system, they can execute any commands and copy any files normally accessible to the uucp login. It is up to the individual sites to be aware of this and apply the protections that they feel are necessary to prevent unauthorized use of files and commands.

The uucp system does provide a certain level of security. For example, a calling site does not get a standard shell when it logs in. Instead, the uucico program is started and all work is done through this program. The uucico program checks the pathnames of files to be sent or received to prevent access to restricted directories. The *USERFILE* supplies the information for these checks. To prevent execution of possibly damaging commands, the uuxqt program can only execute the rmail program on a remote site. This special program is one of many underlying mail programs that help deliver mail. Finally, the *L.sys* file is owned by uucp and has mode 0400 to protect the phone numbers and login information for remote sites.

CREATING A NEW DIAL PROGRAM

The dial program is used by dial-out sites to place calls to other computers. You can create a new dial program for any dial-out site not using the Hayes Smartmodem by modifying the source program and compiling it with the cc command. The **dial** source program is included on the following pages.

```
/*
 *
 *      Copyright (C) Microsoft Corporation, 1983
 *
 *  Simple dialer program for the Hayes "Smart" Modem 1200
 *
 *      See Hayes Reference Manual for command definitions
 *
 *  Usage:  dial ttyname telnumber speed
 *
 *      returns 0 if a connection was made
 *              -1 otherwise
 */

#include <stdio.h>
#include <signal.h>
#include <fcntl.h>
#include <sys/types.h>
#include <sys/ioctl.h>
#include <termio.h>

#define SAME    0
char *setup = "M1 F1 DT";          /* Speaker on, Full Duplex, Touch tone*/
struct termio term;
int baudrate;                           /* baud rate of modem */
char buffer[80];
int alrmint();

main(argc,argv)
int argc;
char *argv[];
{
   FILE *fdr,*fdw;
   int fd;

   if( argc != 4) {
       fprintf(stderr,"Usage: dial  devicename [number] speed\n");
       exit(-1);
   }
   if( (fd=open(argv[1],O_ROWR|O_NDELAY)) < 0 ) {
       fprintf(stderr,"dial: Can't open device: %s for reading.\n",argv[1]);
       exit(-1);
   }
   switch(atoi(argv[3])) {
       case 300:
                   baudrate = B300;
                   break;
           case 1200:
                   baudrate = B1200;
                   break;
           default:
                   baudrate = B1200;
   }
   /*
    * set line for no echo and specific speed
```

```
     */
    ioctl(fd, TCGETA, &term);
    term.c_cflag &= ~CBAUD;
    term.c_cflag |= CLOCAL|HUPCL|baudrate;
    term.c_lflag &= ~ECHO;
    term.c_cc[VMIN] = 1;
    term.c_cc[VTIME] = 0;
    ioctl(fd, TCSETA, &term);
    fcntl(fd, F_SETFL, fcntl(fd, F_GETFL, 0) & ~O_NDELAY);
    if( (fdr=fopen(argv[1],"r")) == NULL ) {
        fprintf(stderr,"dial: Can't open device: %s for reading.\n",argv[1]);
        exit(-1);
    }
    if( (fdw=fopen(argv[1],"w")) == NULL ) {
        fprintf(stderr,"dial: Can't open device: %s for writing.\n",argv[1]);
        exit(-1);
    }
    setbuf(fdw,0);                          /* Want unbuffered I/O */
    /*
     * setup for timeout in 10 seconds if no response
     */
    signal(SIGALRM, alrmint);
    alarm(10);
reread:
    fprintf(fdw,"AT\r");                /* Put Hayes into command mode */
    if( fgets(buffer,sizeof buffer, fdr) == NULL )
            exit(-1);
    if( strncmp(buffer, "OK",2) != SAME ) {         /* got back an OK? */
            sleep(1);
            goto reread;
    }
    alarm(0);                                       /* turn off alarm */
    sleep(1);
    fprintf(fdw,"AT %s %s\r",setup,argv[2]);       /* put out dialing string */
    /*
     * turn off CLOCAL now, since we want modem interrupts to work
     * setup alarm.  (Longer timeout period for longer numbers)
     *
    ioctl(fd, TCGETA, &term);
    term.c_cflag &= ~CLOCAL;
    ioctl(fd, TCSETA, &term);
    */
    alarm((4*strlen(argv[2])) + 5);
again:
    if( fgets(buffer,sizeof buffer,fdr) == NULL )
                    exit(-1);
    if( strncmp(buffer, "NO CARRIER",10) == SAME ) {
            exit(-1);
    }
    if( strncmp(buffer, "CONNECT",7) != SAME ) {
            goto again;
    }
    exit(0);
}
    alrmint()
    {
       exit(-1);
    }
```

A. XENIX SPECIAL DEVICE FILES

ABOUT THIS APPENDIX

This appendix summarizes the various special device files.

CONTENTS

INTRODUCTION

This appendix contains the information you need to create file systems and add terminals to the XENIX system. For a full description of the special files mentioned here, see Section (M) of the XENIX User and System Administrator Reference Manual.

FILE SYSTEM REQUIREMENTS

Many of the file system maintenance tasks described in this guide require the use of special filenames, block sizes, and gap and block numbers. The following sections describe each in detail.

SPECIAL FILENAMES

A special filename is the name of the device special block or character I/O file that corresponds to a peripheral device, such as a hard or diskette drive. These names are required in such commands as mkfs , mount and df to specify the device that contains the file system you are creating, mounting, or searching.

The following table lists the special block I/O filenames and corresponding devices for the diskette drive(s) on your M28.

SPECIAL FILENAME	DISKETTE DRIVE	DENSITY (tpi)	SECTORS	SIDES	CAPACITY (Kbytes)
/dev/fd0	Drive 0	(default for drive 0, associated with /dev/fd096ds15)			
/dev/fd1	Drive 1	(default for drive 1, associated with /dev/fd148ds9)			
/dev/fd048ss8	Drive 0	48	8	1	160
/dev/fd048ds8	Drive 0	48	8	2	320
/dev/fd048ss9	Drive 0	48	9	1	180
/dev/fd048ds9	Drive 0	48	9	2	360
/dev/fd096ds1	Drive 0	96	15	2	1200
/dev/fd148ss8	Drive 1	48	8	1	160
/dev/fd148ds8	Drive 1	48	8	2	320
/dev/fd148ss9	Drive 1	48	9	1	180
/dev/fd148ds9	Drive 1	48	9	2	360
/dev/fd196ds1	Drive 1	96	15	2	1200

For each block I/O entry in the above table, there is an equivalent character I/O or raw device filename. A raw filename is formed by prepending an "r" to the device name. Thus /dev/rfd0, /dev/rfd1 and so on. Raw device filenames are used by the format command and may also be used by certain applications.

The following table lists the special block I/O filenames and corresponding partition names for the hard disk(s) in your M28 system:

SPECIAL FILENAME	HARD DISK DRIVE	PARTITION
/dev/root	Drive 0	The root partition (equivalent to /dev/hd03)
/dev/usr	Drive 0	The user partition (equivalent to /dev/hd02)
/dev/hd00	Drive 0	All partitions
/dev/hd01	Drive 0	4th partition
/dev/hd02	Drive 0	3rd partition (/dev/usr)
/dev/hd03	Drive 0	2nd partition (/dev/root)
/dev/hd04	Drive 0	1st partition (bad track table)
/dev/hd10	Drive 1	All partitions
/dev/hd11	Drive 1	4th partition
/dev/hd12	Drive 1	3rd partition
/dev/hd13	Drive 1	2nd partition
/dev/hd14	Drive 1	1st partition (bad track table)

Note that the device names are associated in reverse order with the partition numbers assigned during the installation procedure and by fdisk . Because of this anomaly you are advised whenever possible to use the names /dev/root and /dev/usr instead of the numeric equivalents.

As with diskettes, there is a character I/O or raw device name for every block I/O filename. Again, this is formed by prepending an "r" to the filename. Thus /dev/rhd00, /dev/rhd10 and so on. These raw device names are used by the badblock and fdisk commands and by certain applications.

The special device filenames for the optional cartridge tape drive are as follows:

SPECIAL FILENAME	MEANING
/dev/rtpnv	The tape drive (with verification performed at backup time)
/dev/rtpnn	The tape drive (without verification performed)

BLOCK SIZES

The block size of a disk is the number of blocks of storage space that is available on the disk. Many commands require input that defines the number of blocks to be operated on. Other commands report disk space in terms of blocks.

Block sizes are sometimes measured in 512-byte units and sometimes in 1024-byte (1K) units. In general it can be said that while in XENIX physical sectors are only 512 bytes long, file systems are measured in 1024-byte blocks. The commands that you need to use to create hard disk partitions and make file systems, fdisk and mkfs , both require input expressed in 1024-byte blocks.

The 1024-byte block size of a diskette depends on the total storage capacity of the diskette as given by the manufacturer, and the way in which it has been formatted. The following table shows possible block sizes for diskettes suitable for use in M28 diskette drives:

Diskette Format	Block Size
8 sector, single-sided	160
8 sector, double-sided	320
9 sector, single-sided	180
9 sector, double-sided	360
15 sector, double-sided	1200

The default value is 1200 for high capacity diskettes formatted in drive 0 with the format command.

The block size of a hard disk partition depends on the size of the partition created at installation time or by means of the fdisk command. You can estimate the block size of a partition if you have the following information:

- The total capacity of the hard disk (for example 20 Mbytes or 40 Mbytes). This information should be readily available to you when you acquire your system.
- The total number of cylinders available (this information should also be available when you acquire your system, and is indicated when you select the create option of the fdisk command).
- The number of cylinders used by the partition in question (this information can be arrived at using the display option of fdisk).

The block size equals:

$$\frac{\text{capacity (in bytes)}}{1024} \times \frac{\text{cylinders used}}{\text{total cylinders}}$$

GAP AND BLOCK NUMBERS

The gap and block numbers are used by the mkfs command to describe how the blocks will be arranged on a disk. The following table lists the gap and block numbers for the disks used with the M2B.

Disks	Gap	Block
9 sectors diskette	3	18
15 sectors diskette	3	30
Hard disk	1	34

TERMINAL AND NETWORK REQUIREMENTS

When you use the enable and disable commands to add and remove terminals on a system, or the install option of the netutil program to build a network, you will need to provide the name of the serial lines that will connect a terminal or network. The following table lists the device special filenames of the serial lines used with the M2B.

Filename	Line
/dev/console	Serial line for the system console
/dev/tty00	Serial line 0
/dev/tty01	Serial line 1

B. XENIX DIRECTORIES

ABOUT THIS APPENDIX

This appendix summarizes the main XENIX directories.

CONTENTS

INTRODUCTION

This appendix lists the most frequently used files and directories in the XENIX system. Many of these files and directories are required for proper XENIX operation and must not be removed or modified. The following sections briefly describe each directory.

THE ROOT DIRECTORY

The root directory (/) contains the following system directories:

/bin	XENIX command directory
/dev	Device special directory
/etc	Additional program and data file directory
/lib	C program library directory
/mnt	Mount directory (reserved for mounted file systems)
/usr	User home directories
/tmp	Temporary directory (reserved for temporary files created by programs)

All of the above directories are required for system operation.

The root directory also contains a few ordinary files. Of these files, the most notable is the xenix file which contains the XENIX kernel image.

THE /bin DIRECTORY

The /bin directory contains the most common XENIX commands, that is, the commands likely to be used by anyone on the system. The following is a list of a few of the commands:

basename	echo	passwd	su
cp	expr	rm	sync
date	fsck	sh	tar
backup	login	sleep	restore
dumpdir	mv	stty	test

These commands and all others in the /bin directory are required for the XENIX operating system and must not be removed.

THE /dev DIRECTORY

The /dev directory contains special device files that control access to peripheral devices. All files in this directory are required and must not be removed. The following is a sample list of the files:

/dev/console	System console
/dev/lp	Lineprinter
/dev/mem	Physical memory
/dev/null	Null device (used to redirect unwanted output)
/dev/rXX	Unbuffered interface to corresponding device name
/dev/root	Root file structure
/dev/swap	Swap area
/dev/ttyXX	Terminals
/dev/tty	The terminal you are using

THE /etc DIRECTORY

The /etc directory contains miscellaneous system program and data files. All files are required, but many may be modified.

The following program and data files must not be removed or modified.

/etc/mtab	Mounted device table
/etc/mount	For mounting a file structure
/etc/mkfs	For creating a file structure
/etc/init	First process after boot

The following data files are amongst those that may be modified, if desired. No file may be removed.

/etc/passwd	Password file
/etc/rc	Bootup shell script
/etc/ttys	Terminal set up
/etc/termcap	Terminal capability map
/etc/motd	Message of the day.
/etc/profile	Profile file

THE /lib DIRECTORY

The /lib directory contains runtime library files for C and other language programs. The directory is required and must not be removed.

THE /mnt DIRECTORY

The /mnt directory is an empty directory reserved for mounting removable file systems.

THE /tmp DIRECTORY

The /tmp directory contains temporary files created by XENIX programs. The files are normally present when the corresponding program is running, but may also be left in the directory if the program is prematurely stopped. You may remove any temporary file that does not belong to a running program.

The /usr DIRECTORY

The */usr* directory contains the home directories of all users on the system. It also contains several other directories that provide additional XENIX commands and data files.

The /usr/bin directory contains more XENIX commands. These commands are less frequently used or considered nonessential to XENIX system operation.

The /usr/include directory contains header files for compiling C programs.

The /usr/lib directory contains more libraries and data files used by various XENIX commands.

The /usr/spool directory contains various directories for storing files to be printed, mailed, or passed through networks.

The /usr/tmp directory contains more temporary files.

The /usr/adm directory contains data files associated with system administration and accounting. In particular, the /usr/adm/messages file contains a record of all error messages sent to the system console. This file is especially useful for locating hardware problems. For example, an unusual number of disk errors on a drive indicates a defective or misaligned drive. Since messages in the file can accumulate rapidly, the file must be deleted periodically.

LOG FILES

A variety of directories contain log files that grow in size during the normal course of system operation. Many of these files must be periodically cleared to prevent them from taking up valuable disk space (see Chapter 5, "Maintaining File Systems"). The following table lists the files (by full pathname) and their contents:

FILENAME	DESCRIPTION
/etc/ddate	Records date of each backup.
/usr/adm/pacct	Records accounting information; grows rapidly when processs accounting is on.
/usr/adm/messages	Records error messages generated by the system when started.
/usr/adm/wtmp	Records user logins and logouts.
/usr/adm/sulog	Records each use of the su command; grows only if the option is set in the /etc/default/su file.
/usr/spool/at/past	Records each use of the at command.
/usr/spool/micnet/*/LOG	Records transmissions between machines in a Micnet network. The * must be the name of a remote machine connected to the current machine.
/usr/lib/cron/log	Records cron executions.

C. XENIX CONFIGURATION FILES

ABOUT THIS APPENDIX

This appendix lists the configuration files provided with your system.

CONTENTS

INTRODUCTION

This appendix lists and describes the configuration files supplied by Olivetti as part of the XENIX V distribution. See Chapter 9 for full instructions on how to install a configuration file.

CONFIGURATION FILES

CONFIGURATION FILE	HAROWARE SUPPORTED
config00.sys	single-user with diskette drive
config01.sys	single-user with diskette drive and parallel printer
config02.sys	single-user with diskette drive and tape drive
config03.sys	single-user with diskette drive, parallel printer and tape drive
config04.sys	single-user with diskette drive, parallel printer and serial printer
config05sys	single-user with diskette drive, parallel printer, serial printer and tape drive
config10.sys	multi-user with diskette drive, parallel printer and US version WS584/VT100
config11.sys	multi-user with diskette drive, parallel printer, tape drive and US version WS584/VT100
config12.sys	multi-user with diskette drive, parallel printer, serial printer and US version WS584/VT100
config13.sys	multi-user with diskette drive, parallel printer, serial printer, tape drive and US version WS584/VT100
config20.sys	multi-user with diskette drive, parallel printer and German version WS584/VT100
config21.sys	multi-user with diskette drive, parallel printer, tape drive and German version WS584/VT100
config22.sys	multi-user with diskette drive, parallel printer, serial printer and German version WS584/VT100
config23.sys	multi-user with diskette drive, parallel printer, serial printer, tape drive and German version WS584/VT100

config30.sys	multi-user with diskette drive, parallel printer and French version WS584/VT100
config31.sys	multi-user with diskette drive, parallel printer, tape drive and French version WS584/VT100
config32.sys	multi-user with diskette drive, parallel printer, serial printer and French version WS584/VT100
config33.sys	multi-user with diskette drive, parallel printer, serial printer, tape drive and French version WS584/VT100
config40.sys	multi-user with diskette drive, parallel printer and Spanish version WS584/VT100
config41.sys	multi-user with diskette drive, parallel printer, tape drive and Spanish version WS584/VT100
config42.sys	multi-user with diskette drive, parallel printer, serial printer and Spanish version WS584/VT100
config43.sys	multi-user with diskette drive, parallel printer, serial printer, tape drive and Spanish version WS584/VT100
config50.sys	multi-user with diskette drive, parallel printer and UK version WS584/VT100
config51.sys	multi-user with diskette drive, parallel printer, tape drive and UK version WS584/VT100
config52.sys	multi-user with diskette drive, parallel printer, serial printer and UK version WS584/VT100
config53.sys	multi-user with diskette drive, parallel printer, serial printer, tape drive and UK version WS584/VT100
config60.sys	multi-user with diskette drive, parallel printer and Swedish/ Finnish version WS584/VT100
config61.sys	multi-user with diskette drive, parallel printer, tape drive and Swedish/Finnish version WS584/VT100
config62.sys	multi-user with diskette drive, parallel printer, serial printer and Swedish/Finnish version WS584/VT100
config63.sys	multi-user with diskette drive, parallel printer, serial printer, tape drive and Swedish/Finnish version WS584/VT100
config70.sys	multi-user with diskette drive, parallel printer and Norwegian/ Danish version WS584/VT100
config71.sys	multi-user with diskette drive, parallel printer, tape drive and Norwegian/Danish version WS584/VT100

config72.sys	multi-user with diskette drive, parallel printer, serial printer and Norwegian/Danish version WS584/VT100
config73.sys	multi-user with diskette drive, parallel printer, serial printer, tape drive and Norwegian/Danish version WS584/VT100

NOTICE

Ing. C. Olivetti & C., S.p.A. reserves the right to make any changes in the product described in this manual at any time and without notice.

This manual is licensed to the Customer under the conditions contained in the User License enclosed with the Program to which the manual refers.

Code 4022960 S (1)
Printed in Italy

olivetti
PERSONAL
COMPUTER

olivetti