

Plexus Sys5 UNIX S/M/K1.7 Release Notice

98-40199.7 Ver. A October, 1987

Plexus Sys5 UNIX S/M/K1.7 Release Notice

98-40199.7 Ver. A October, 1987

PLEXUS COMPUTERS, INC.

3833 North First Street

San Jose, CA 95134

408/943-9433

Copyright 1987
Plexus Computers, Inc., San Jose, CA

All rights reserved.

No part of this publication may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language, in any form or by any means, without the prior written consent of Plexus Computers, Inc.

The information contained herein is subject to change without notice. Therefore, Plexus Computers, Inc. assumes no responsibility for the accuracy of the information presented in this document beyond its current release date.

Printed in the United States of America

CONTENTS

INTRODUCTION	1-1
1.1 Conventions	1-2
CONTENTS	2-1
2.1 Contents Of Release Tape	2-1
2.2 Contents Of Document	2-2
LOAD S/M/K1.7 ONTO AN EXISTING SYSTEM	3-1
3.1 Back Up The File Systems	3-2
3.1.1 Make cpio Backups	3-2
3.1.2 Make A dump Backup Of The Root File System	3-3
3.2 Perform Hardware Upgrade	3-3
3.3 Organize The Disk	3-3
3.3.1 Initial Configuration	3-4
3.3.2 dconfig Example With Plexus Default Values	3-6
3.3.2.1 dconfig Example For A Disk Controlled By An IMSP	3-7
3.3.2.2 dconfig Example For A Disk Controlled By An EMSP	3-8
3.3.2.3 dconfig Example For A Disk Controlled By An AMSP	3-9
3.3.2.4 dconfig Example For A Disk Controlled by An MMSP	3-10
3.3.3 dconfig Example With The Old Plexus Default Layout	3-11
3.3.4 dconfig Example With Alternate File System As Root	3-12
3.3.5 Reorganize Current Configuration	3-13
3.4 Create The Root File System	3-16
3.4.1 Run mkfs	3-16
3.4.1.1 mkfs Example	3-16
3.4.2 Run fsck	3-17
3.4.2.1 fsck Example	3-18
3.5 Install The Minimum Root File System	3-18
3.5.1 Run restor	3-19
3.5.1.1 restor Example	3-19
3.6 Boot Into Single-User Mode	3-19
3.7 Install The Standard Release Files	3-20
3.8 Install The Manual Pages (optional)	3-20
3.9 Retrieve Files From The Root File System Backup	3-21
3.10 Install The Update File	3-21
3.11 Reioad Backups	3-22

COMPLETE THE INSTALLATION	4-1
4.1 Load The Files Onto Disk	4-2
4.2 Manually Integrate Modified Files	4-3
4.2.1 /etc/bcheckrc	4-3
4.2.2 /etc/brc	4-3
4.2.3 /etc/cshprofile	4-4
4.2.4 /etc/gettydefs	4-4
4.2.5 /etc/inittab	4-4
4.2.6 /etc/ioctl.syscon	4-4
4.2.7 /etc/passwd	4-5
4.2.8 /etc/profile	4-5
4.2.9 /etc/rc	4-6
4.2.10 /etc/TIMEZONE	4-6
4.2.11 /etc/shutdown	4-6
4.2.12 /usr/spool/cron/crontabs Files: adm, root, sys, uucp	4-7
4.2.13 /usr/src/uts/m68/cf/conf.c	4-7
4.3 Manually Integrate Kernel Files	4-8
4.3.1 Ensure Kernel File Integrity	4-8
4.3.2 Integrate Kernel Files With Added Drivers	4-8
4.4 Relink The Kernel	4-9
ENTER MULTI-USER MODE	5-1
5.1 Create Or Modify /etc/inittab	5-1
5.1.1 Create /etc/inittab	5-1
5.1.2 Modify /etc/inittab	5-2
5.1.3 Create /dev Entries	5-3
5.2 Modify /etc/rc	5-4
5.2.1 Modify /etc/rc For S/M1.7	5-4
5.2.2 Modify /etc/rc For K1.7	5-5
5.3 Configure A VCP	5-5
5.4 Verify /dev/swap	5-6
5.5 File Systems Created Prior To Sys5 Version 1.4	5-6
5.6 Boot UNIX From Tape	5-7
5.7 Carrier Detect On ACP	5-7
5.8 Back Up The Root File System	5-7
GENERAL INFORMATION	6-1
6.1 Special Device Names	6-1
6.2 Sys5/Sys3 Device Names	6-2
6.3 Some Modifications To System V	6-3
6.3.1 Not Provided – Not Applicable	6-3
6.3.2 Not Provided – No Source	6-4
6.4 Known Problems	6-4
6.5 Problems Fixed In This Release	6-6
6.5.1 ACP Problems Fixed In S/M/K1.7	6-7

ENHANCEMENTS	7-1
CONFIGURABLE FILES	8-1
8.1 Files In /usr/src/uts/m68/cf	8-1
8.2 Inside conf.c	8-2
PLEXUS SOFTWARE SUPPORT POLICY	9-1
9.1 Warranty Coverage	9-1
9.2 Support Hours	9-1
9.3 Preparation For Requesting Support	9-1

LIST OF TABLES

Table 2-1. Description of Release Tape	2-1
Table 2-2. Standalone Files of the Sys5 Release Tape	2-2
Table 3-1. File System to Reorganize	3-15
Table 3-2. Reorganized File System	3-15
Table 3-3. Gap and Blocks per Cylinder	3-17
Table 6-1. Cross References Between Sys5 and Sys3	6-2



.

.



.

.



Chapter 1

INTRODUCTION

This release notice accompanies Plexus' release of *Sys5 UNIX† S/M/K1.7*, which is an implementation of the AT&T *System V* Version 2.0 operating system by Plexus Computers for the P/35, P/55, P/60, P/75, P/90, and P/95 computer systems. This document is a collection of information about *Sys5 S/M/K1.7*.

NOTE

- If you are simultaneously upgrading from Sys3 to Sys5 AND upgrading from the MC68000‡ to the MC68020, you must first satisfy the hardware requirements listed in the *68020 CPU Installation Manual* (Plexus publication number 98-40184.2).
- If you have a previous version of Sys5 already installed and are upgrading to the MC68020 CPU board, perform the hardware upgrade when instructed to do so in this document.

If you have received a new Plexus P/35, P/55, P/60, P/75, P/90, or P/95 system, *Sys5 S/M/K1.7* was installed by Plexus at the factory. Keep this release notice as a reference. You will not have to do any further installation at this time, although you will probably need to perform site-specific customizations as described in the chapter: *COMPLETE THE INSTALLATION*.

As part of the software installation, a *System Format and Load* listing is generated and sent with your system. This listing is a hardcopy of the procedures that installed your software. **SAVE THIS LIST**; it contains information you might need if you experience a catastrophic system failure. Refer to the user's manual that you received with your system, or the *Sys5 Administrator's Handbook* for start up and shut down procedures.

Use this release notice as a supplement to the user's manual for your system or the *Sys5 Administrator's Handbook* (Plexus publication number 98-05133.2), and the Installation Guide that were shipped with your Plexus system.

† UNIX is a trademark of AT&T Bell Laboratories, Inc.

‡ MC68000 and MC68020 are trademarks of Motorola, Inc.

1.1 Conventions

This document observes the following conventions for consistency and accuracy of communication.

- *System V* refers to the standard software provided by AT&T. *Sys5* refers to Plexus' implementation of this software.
- *Sys5* and UNIX commands, utilities, programs, files, etc. are identified by boldface type (e.g., **getty**).

Numbers in parentheses after a command name—such as **dnld(1)**—indicate the section of the *Plexus Sys5 UNIX Reference Manual* where that command is documented.

- Operator entry instructions are set off from the rest of the text and are in bold face.

A carriage return is to be pressed after each command line unless instructed otherwise.

If it seems necessary to indicate a carriage return, this will be done with **<cr>**.

Arguments and variables which must be replaced with actual filenames and constants are in *italics*. For example, in the example **/dev/dsk/xs0**, the *x* is a variable and will be explained following the instruction.

- System messages and responses are depicted in a special **constant width** font. In some cases these are provided as examples of typical responses, and may not be identical to the responses that you see during the installation.

Chapter 2

CONTENTS

This release of *Plexus Sys5 UNIX S/M/K1.7* includes the following:

- A release tape, which may be a one cartridge version, two cartridge version, or reel-to-reel.
- This release notice, Plexus publication number 98-40199.7.

We recommend that you also purchase the *Sys5 Administrator's Handbook*, Plexus publication number 98-05133.2.

2.1 Contents Of Release Tape

Table 2-1. Description of Release Tape

File Number(s)	Position to Start	Bytes per Block	Format	Description
0-19	n/a	1024	binary dump	Standalone programs (see Table 2-2)*
20	20	10240	binary dump	Bootstrap minimum file system release for <i>Sys5</i> and configurable files.
21	21	5120	cpio	Files comprising the full release except /usr/catman and configurable files.
22	22	5120	cpio	Files comprising /usr/catman .
23	23	5120	cpio	Configurable files requiring some manual integration.
24	24	5120	cpio	Updates to software.

n/a = not applicable

- * File numbers 0 – 19 are for backup and emergency purposes in case the disk copies of the standalones become inaccessible and you need to run them from a tape.

☞ NOTE ☞

- For the two-cartridge version, both tapes have 25 files. The first tape has information in all files except file 21, which is empty. The second tape has information only in file 21. The rest of the files are empty.
 - When instructed to insert the release tape into the tape drive, use the first volume unless instructed otherwise.
-

Table 2-2 lists the standalone files (file numbers 0 – 19) of the Sys5 Release Tape in order.

Table 2-2. Standalone Files of the Sys5 Release Tape

File #	Programs	File #	Programs
0	help	1	dummy
2	unix	3	dformat
4	mkfs	5	restor
6	fsck	7	dd
8	fbackup*	9	od
10	dconfig	11	dummy
12	fsdb	13	du
14	ls	15	cat
16	badtrack*	17	dummy
18	dummy	19	dummy

* Correct for S/M1.7 only. For K1.7 these are both dummy files.

2.2 Contents Of Document

This document contains the following chapters:

- Chapter 1 contains information to help you use this document.
- Chapter 2 describes the contents of this release.
- Chapter 3 describes how to upgrade from an earlier version of software to S/M/K1.7.
- Chapter 4 describes how to install S/M/K1.7 onto a new machine, and continues instructions for installation onto an existing system.
- Chapter 5 describes site-dependent procedures necessary to bring the system into multi-user mode.
- Chapter 6 contains general information about Plexus Sys5.
- Chapter 7 describes new features of this release of Sys5.
- Chapter 8 describes the files in the directory `/usr/src/uts/m68/cf`, with special emphasis on the file `conf.c`.
- Chapter 9 describes the Plexus Software Support Policy.

LOAD S/M/K1.7 ONTO AN EXISTING SYSTEM

If you have just received your machine from Plexus with 1.7 installed on it, the contents of files 20, 21, and 24 will already be on the disk. You should proceed to the chapter *COMPLETE THE INSTALLATION*, which outlines simple customization steps used to bring a machine to full operational capability.

If you have had a catastrophic failure for which no usable backup exists, or if the software on the machine is from a release previous to 1.7, you will need to follow the procedures in this chapter.

 **NOTE** 

You must not have the autoboot switch turned on when installing new software. Please refer to the *Sys5 Administrator's Handbook* or the user's manual for your system for information on how to disable this switch if it is turned on.

3.1 Back Up The File Systems

If there has not been a catastrophic failure, you will first want to back up the file systems. Make **cpio** backups of all of the file systems except the root file system. The root file system should be backed up using **dump**, so that, if necessary, it can be reloaded in standalone mode using **restor**.

3.1.1 Make cpio Backups

- This backup must be done in single-user mode.

Shut down the system as you normally would, and make sure that all of the file systems are unmounted. To display a list of mounted devices, enter **mount**. Use the **umount** command to unmount them.

To make **cpio** backups of your file systems follow the steps given below, for each file system, in turn.

1. From the root directory mount the file system onto the "dummy" mount stub **/mnt**. Enter:

```
/etc/mount xxxx /mnt
```

Where: *xxxx* is the logical file system device name (for example, **/dev/dsk/0s3** or **/dev/dk3**).

2. Change into the directory **/mnt** and write the files to tape. Insert a blank tape into the tape drive and enter:

```
cd /mnt  
find . -cpio /dev/xxx/0m
```

Where: *xxx* is the name of your default tape drive
rpt for cartridge drives
rrm for reel-to-reel drives,
crm for caching reel-to-reel drives.

If your **cpio** backup needs more than one tape, at the end of the tape it will display the message:

```
Cannot write to tape, enter device name to continue.
```

Remove the full tape and label it with the file system name and tape volume number. Insert a new tape and type the name of your tape drive, followed by a carriage return.

3. Return to the top level directory and unmount the file system. Enter:

```
cd /  
/etc/umount xxxx
```

Where: *xxxx* is the logical file system device name (as used in Step 1).

3.1.2 Make A dump Backup Of The Root File System

- This backup must be done in single-user mode.

Insert a blank tape into the tape drive and enter one of the following:

- If you have a reel-to-reel tape drive, enter:

```
/etc/dump Ou xxxx
```

Where: *xxxx* is the logical file system device name of your root file system. For example, **/dev/dsk/0s1**.

- If you have a cartridge drive, enter:

```
/etc/dump Ousf ssss zzzz xxxx
```

Where: *ssss* is **1000** for 20Mb cartridge tape drives,
ssss is **2000** for 45Mb cartridge tape drives,
ssss is **2500** for 60Mb cartridge tape drives,
zzzz is the name of your tape drive, (such as **/dev/rpt/0m** or **/dev/rpt0**),
xxxx is the logical file system device name of your root file system (such as **/dev/dsk/0s1**).

The **/etc/dump** file will report on the estimated number of blocks it will back up, and the estimated number of tapes it will use.

If **dump** needs more than one tape, it will prompt you with the message:

```
change tapes
```

Remove the full tape from the drive and label it with the file system name and the volume number. Place a new tape in the drive and press carriage return.

3.2 Perform Hardware Upgrade

If you are upgrading your system from an MC68000 CPU to an MC68020 CPU, perform the hardware upgrade at this time. This procedure is described in the *68020 CPU Installation Manual*, Plexus publication number 9E-40184.2.

3.3 Organize The Disk

The space requirement for the root file system for UNIX Sys5 S/M/K1.7 is at least 34Mb of file space. If your root file system is not large enough, you will need to enlarge it. To do this, you will probably have to remove space from at least one other file system. To check the current disk configuration, run the standalone **dconfig**, following the procedure described in the next two sections.

3.3.1 Initial Configuration

This section describes using **dconfig** with a new machine, or with a machine that has had a catastrophic failure.

The Plexus disk configuration program used with the P/35, P/55, P/60, P/75, P/90 and P/95 systems is **dconfig**. It can be used to describe both the physical and logical attributes of a disk. The physical attributes are customarily set up using **dformat**, however, and this document assumes that you have already formatted the disk. If this is not true, and the disk is brand-new out of the carton, you must format it. See the user's manual for your system or the *Sys5 Administrator's Handbook* for information on this procedure.

The logical attributes of the disk that you may be altering are the default boot program, the system node name, the logical devices to use for **root**, **pipe**, **swap** and **dump** procedures, the location and size of the disk swap space, and the logical file system configuration.

Please note that the minimum recommended size for the root file system has been increased to 34Mb. You will need 23Mb to install the release, plus 5Mb if you wish to install the manuals. The extra space is needed as temporary storage for the various programs that use **/tmp**. If you wish to mount **/tmp** as a separate file system, you can make the root file system somewhat smaller than the recommended size.

Examples of **dconfig** sessions are provided in the next subsection. You may want to refer to them while reading the following description of the actions you will be performing.

The default boot program name should be **/unix**. If you are using a file system for your root file system that does not begin at the first block of the first physical disk, you must enter an offset before the boot name. The name will be of the form:

`disktype(#,#)/unix`

Where: *disktype* is the type of controller (**pd** for IMSP, **xy** for EMSP, **sm** for AMSP, and **sd** for MMSP)
 the first number (#) is the number of the physical disk,
 the second number is the sector offset where the logical file system you wish to use as root is located.

For example, if you wish to use a disk controlled by an IMSP that is your system's first physical disk, and you wish to use the third logical file system that begins at sector 80000 for your root file system, your default boot name would be **pd(0,80000)/unix**. If you wish to use an EMSP controlled disk that is your system's second physical disk, and you wish to use the second logical file system that begins at sector 60000 for your root file system, your default boot name would be **xy(1,60000)/unix**. See the section *dconfig Example With The Alternate File System Used As Root* for an example of this.

The system node name should be a name unique within the group of systems your machine will be communicating with. This node name is used by **uucp**, TCP/IP, and **NOS** to identify a specific machine. The system node name can be up to six characters long.

The logical devices to use for **root** and **pipe** should be the same number. This number, which is the index number of a logical file system, determines which device is opened to use as the root file system. For example, if you use the first logical file system on the disk as your root file system, **rootdev** and **pipdev** are set to 1. If you use the tenth logical file system as your root file system, **rootdev** and **pipdev** are set to 10.

The logical devices to use for **swap** and **dump** should also be the same number. This number, again an index number of a logical file system, determines which device is opened to use as the swap device. For example, if you use the second logical file system on the disk as your swap space, **dumpdev** and **swapdev** are set to 2.

In addition to providing a device number for **swapdev**, you must set up the starting sector number and the sector size of the **swap** device. These two numbers are prompted for with **swplo** (the starting sector number) and **nswap** (the sector size of the swap space). If the swap space is on a logical file system that will be used only for swapping (the recommended layout), **swplo** will be 0. If the swap space is on a logical file system that will have an active file system on the first part and the swap space at the end, **swplo** + **nswap** must equal the sector size of the logical file system. The block count to use when you create the file system using **mkfs** must be **swplo** divided by 2.

The first diagram shows a layout using separate file systems for **root** and **swap**. In this example, **dumpdev** and **swapdev** would be set to 2, **swplo** would be set to 0, and **nswap** would be set to 32000. The sector count set up for **/dev/dsk/0s1** (as described below) would be 68000, the sector count for **/dev/dsk/0s2** would be 32000. When an **mkfs** is run for **/dev/dsk/0s1**, the size used would be 34000 (68000/2). The following section, *dconfig Example With Plexus Default Values*, gives examples of this.

```

sector 0          68000          100000          end of disk
|          root          |          swap          |          rest of file systems          |
-----
[ /dev/dsk/0s1 ] [ /dev/dsk/0s2 ] [ /dev/dsk/0s3 - /dev/dsk/0s15 ]

```

The second diagram shows a layout using the same file system for **root** and **swap**. In this example, **dumpdev** and **swapdev** would be set to 1, **swplo** would be set to 68000 and **nswap** would be set to 32000. The sector count set up for **/dev/dsk/0s1** (as described below) would be 100000. When an **mkfs** is run for **/dev/dsk/0s1**, the size used would again be 34000 (68000/2). If a larger size were used, the file system would become corrupt when swapping occurred. The section *dconfig Example With The Old Plexus Default Layout* gives an example of this.

```

sector 0          68000          100000          end of disk
|          root          |          swap          |          rest of file systems          |
-----
[          /dev/dsk/0s1          ] [ /dev/dsk/0s2 - /dev/dsk/0s15 ]

```

nswap depends on how many users your system will have and what they will be doing. A system with few users or light usage will need a smaller swap space than a system with many users or heavy usage, such as a lot of program compilation. Refer to your system's user's manual or the *Sys5 Administrator's Handbook* if you need advice on how to determine the needed amount of swap space. A rough rule of thumb for this is 4Mb (8000 sectors) as a minimum, based on an eight user load with light usage. Add another 2Mb (4000 sectors) per eight users for light loads, 4Mb (8000 sectors) per eight users for heavy loads. Thus, a thirty-two user system with heavy loads should work well with 16Mb (32000 sectors) of swap space.

Please note that if you wish to use a logical file system other than **/dev/dsk/0s2** for your swap space, you must change the definition of **/dev/swap** after you install your software (see the section *Verify /dev/swap* in the chapter *ENTER MULTI-USER MODE*).

The values you use for the other logical file systems depend on how you want your disk configured. The examples given do not alter these file system descriptions, but you will probably want to set them up at this time.

☞ NOTE ☜

When running **dconfig**, make a note of the sectors per track value of your disk. You will need this information later.

Refer to your system's user's manual or the *Sys5 Administrator's Handbook* for a more complete description on the program **dconfig**.

3.3.2 dconfig Example With Plexus Default Values

The Plexus default disk layout uses **/dev/dsk/0s1** as the root file system, with 34Mb of storage. **/dev/dsk/0s2** is used as the swap space with 16Mb of storage, and the rest of the file system is left for you to configure. After the disk is formatted, all of the logical file system values are zeros. The following examples show how to set up a disk in the Plexus default configuration. An example is given for each disk type, IMSP controlled (**pd**), EMSP controlled (**xy**), AMSP controlled (**sm**), and MMSP controlled disks (**sd—P/90**).

To run **dconfig**, insert the release tape into the tape drive. At the boot prompt (:) enter:

```
dconfig
```

Information about your logical file system(s) and the size of your disk(s) will display (for example, ": sd/0s0 259 MB", and so on, for each disk). The program will then load and return the following prompt:

```
$$ dconfig
```

Enter the physical disk name in the form:

```
/dev/dsk/xs0,
```

Where: *x* = the physical disk number.

You will be prompted for each value in turn. The current value is shown in square brackets. If you wish to retain the value shown, enter a carriage return. The values you should enter to set up the disk are shown in bold type.

Note that using the tilde (~) as sector count means "from the beginning sector to the end of the disk".

3.3.2.1 dconfig Example For A Disk Controlled By An IMSP

At the boot prompt, enter:

dconfig

Information about your logical file system(s) and disk sizes will display and the program will continue:

```

$$ dconfig /dev/dsk/0s0 <cr>
Disk id? [pd]: <cr>
Primary bootname? [/unix]: <cr>
Secondary bootname? [ ]: <cr>
Number of cylinders? [1024]: <cr>
Number of heads on removable? [0]: <cr>
Number of heads on fixed? [16]: <cr>
Data bytes per sector? [512]: <cr>
Sectors per track? [34]: <cr>
Number of alternate cylinders? [20]: <cr>
File system blocksize? [1024]: <cr>
Sys5 Nodename? [ ]: <6 character system node name cr>
Change the default unix device mapping? [y/n]: y <cr>
Rootdev? [0x0]: 1 <cr>
Pipedev? [0x0]: 1 <cr>
Dumpdev? [0x0]: 2 <cr>
Swapdev? [0x0]: 2 <cr>
Swplo? [0]: <cr>
Nswap? [0]: 32000 <cr>
Change the file system disk configuration? [y/n]: y <cr>
File system logical configuration: [sector start,sector count]
  dsk/?s0 [ 0, 0]: 0,~ <cr>
  dsk/?s1 [ 0, 0]: 0,68000 <cr>
  dsk/?s2 [ 0, 0]: 68000,32000<cr>
  dsk/?s3 [ 0, 0]: <cr> *
  dsk/?s4 [ 0, 0]: <cr>
  dsk/?s5 [ 0, 0]: <cr>
  dsk/?s6 [ 0, 0]: <cr>
  dsk/?s7 [ 0, 0]: <cr>
  dsk/?s8 [ 0, 0]: <cr>
  dsk/?s9 [ 0, 0]: <cr>
  dsk/?s10 [ 0, 0]: <cr>
  dsk/?s11 [ 0, 0]: <cr>
  dsk/?s12 [ 0, 0]: <cr>
  dsk/?s13 [ 0, 0]: <cr>
  dsk/?s14 [ 0, 0]: <cr>
  dsk/?s15 [ 0, 0]: <cr>
Is the above information correct? [y/n]: y <cr>
Are you sure you want to rewrite block 0? [y/n]: y <cr>

```

* At dsk/?s3 you can type a "q" to quit from this list.

3.3.2.2 dconfig Example For A Disk Controlled By An EMSP

At the boot prompt, enter:

dconfig

Information about your logical file system(s) and disk sizes will display and the program will continue:

```

$$ dconfig /dev/dsk/0s0 <cr>
Disk id? [xy]: <cr>
Primary bootname? [/unix]: <cr>
Secondary bootname? [ ]: <cr>
Disk type: 0
Number of cylinders? [760]: <cr>
Number of heads on fixed? [19]: <cr>
Sectors per track not including spares? [58]: <cr>
Number of spare sectors? [2]: <cr>
Nodename? [ ]: <6 character system node name cr>
Change the default unix device mapping? [y/n]: y <cr>
Rootdev? [0x0]: 1 <cr>
Pipedev? [0x0]: 1 <cr>
Dumpdev? [0x0]: 2 <cr>
Swapdev? [0x0]: 2 <cr>
Swplo? [0]: <cr>
Nswap? [0]: 32000 <cr>
Change the file system disk configuration? [y/n]: y <cr>
File system logical configuration: [sector start,sector count]
  dsk/?s0   [    0,      0]: 0,~ <cr>
  dsk/?s1   [    0,      0]: 0,68000 <cr>
  dsk/?s2   [    0,      0]: 68000,32000 <cr>
  dsk/?s3   [    0,      0]: <cr> *
  dsk/?s4   [    0,      0]: <cr>
  dsk/?s5   [    0,      0]: <cr>
  dsk/?s6   [    0,      0]: <cr>
  dsk/?s7   [    0,      0]: <cr>
  dsk/?s8   [    0,      0]: <cr>
  dsk/?s9   [    0,      0]: <cr>
  dsk/?s10  [    0,      0]: <cr>
  dsk/?s11  [    0,      0]: <cr>
  dsk/?s12  [    0,      0]: <cr>
  dsk/?s13  [    0,      0]: <cr>
  dsk/?s14  [    0,      0]: <cr>
  dsk/?s15  [    0,      0]: <cr>
Is the above information correct? [y/n]: y <cr>
Are you sure you want to rewrite block 0? [y/n]: y <cr>

```

* At dsk/?s3 you can type a "q" to quit from this list.

3.3.2.3 dconfig Example For A Disk Controlled By An AMSP

At the boot prompt, enter:

dconfig

Information about your logical file system(s) and disk sizes will display and the program will continue:

```

$$ dconfig /dev/dsk/0s0 <cr>
Disk id? [sm]: <cr>
Primary bootname? [/unix]: <cr>
Secondary bootname? [ ]: <cr>
Disk type: 0
Number of cylinders? [823]: <cr>
Number of heads on fixed? [10]: <cr>
Sectors per track not including spares? [63]: <cr>
Number of spare sectors 1
Number of alternate tracks? [50]: <cr>
Nodename? [ ]: <6 character node name cr>
Runt sector enable? [0]: <cr>
Gap 2? [12]: <cr>
Change the default unix device mapping? [y/n]: y <cr>
Rootdev? [0x0]: 1 <cr>
Pipedev? [0x0]: 1 <cr>
Dumpdev? [0x0]: 2 <cr>
Swapdev? [0x0]: 2 <cr>
Swplo? [0]: <cr>
Nswap? [0]: 32000 <cr>
Change the file system disk configuration? [y/n]: y <cr>
File system logical configuration: [sector start,sector count]
dsk/?s0      [0,486990] 0,~ <cr>
dsk/?s1      [0, 0]: 0,68000 <cr>
dsk/?s2      [0, 0]: 68000,32000 <cr>
dsk/?s3      [0, 0]: 100000,~ <cr>
dsk/?s4      [0, 0]: 140000,~ <cr>
dsk/?s5      [0, 0]: 180000,~ <cr>
dsk/?s6      [0, 0]: 220000,~ <cr>
dsk/?s7      [0, 0]: 260000,~ <cr>
dsk/?s8      [0, 0]: 300000,~ <cr>
dsk/?s9      [0, 0]: 400000,~ <cr>
dsk/?s10     [0, 0]: 500000,~ <cr>
dsk/?s11     [0, 0]: 600000,~ <cr>
dsk/?s12     [0, 0]: 700000,~ <cr>
dsk/?s13     [0, 0]: 800000,~ <cr>
dsk/?s14     [0, 0]: 900000,~ <cr>
dsk/?s15     [0, 0]: 1000000,~ <cr>
Is the above information correct? [y/n]: y <cr>
Are you sure you want to rewrite block 0? [y/n]: y <cr>

```

3.3.2.4 dconfig Example For A Disk Controlled by An MMSP

Note that the number of cylinders and heads are fixed numbers. If a different number is entered here, the number in the system will not change.

```

dconfig <cr>
: sd/0s0 137 MB host=0 sid=6 lun=0 size=282450
$$ dconfig /dev/dsk/0s0 <cr>
Disk id? [sd]: <cr>
Primary bootname? [/unix]: <cr>
Secondary bootname? [ ]: <cr>
Number of cylinders? [823]: <cr>
Number of heads on fixed? [10]: <cr>
Date bytes per sector? [596]: <cr>
Sectors per track? [35]: <cr>
Number of controller reserved cylinders 3
Number of alternate cylinders? [13]: <cr>
Nodename? [ ]: <6 character node name cr>
Change the default unix device mapping? [y/n]: y <cr>
Rootdev? [0x0]: 1 <cr>
Pipedev? [0x0]: 1 <cr>
Dumpdev? [0x0]: 2 <cr>
Swapdev? [0x0]: 2 <cr>
Swplo? [0]: <cr> <cr>
Nswap? [0]: 32000 <cr>
Change the file system disk configuration? [y/n]: y <cr>
File system logical configuration: [sector start,sector count]
  dsk/?s0 [0, 548060] 0, ~ <cr>
  dsk/?s1 [0, 0]: 0,68000 <cr>
  dsk/?s2 [0, 0]: 68000,32000 <cr>
  dsk/?s3 [0, 0]: 100000, ~ <cr>
  dsk/?s4 [0, 0]: 140000, ~ <cr>
  dsk/?s5 [0, 0]: 180000, ~ <cr>
  dsk/?s6 [0, 0]: 220000, ~ <cr>
  dsk/?s7 [0, 0]: 260000, ~ <cr>
  dsk/?s8 [0, 0]: 300000, ~ <cr>
  dsk/?s9 [0, 0]: 400000, ~ <cr>
  dsk/?s10 [0, 0]: 500000, ~ <cr>
  dsk/?s11 [0, 0]: 600000, ~ <cr>
  dsk/?s12 [0, 0]: 700000, ~ <cr>
  dsk/?s13 [0, 0]: 800000, ~ <cr>
  dsk/?s14 [0, 0]: 900000, ~ <cr>
  dsk/?s15 [0, 0]: 1000000, ~ <cr>

Is the above information correct? [y/n]: y <cr>
Are you sure you want to rewrite block 0? [y/n]: y <cr>

```

3.3.3 dconfig Example With The Old Plexus Default Layout

This layout uses `/dev/dsk/0s1` for both the root file system and the swap space. The sizes used have been increased to reflect the larger size of the release. At the boot prompt, enter:

dconfig

Information about your logical file system(s) and disk sizes will display and the program will continue:

```

$$ dconfig /dev/dsk/0s0 <cr>
Disk id? [pd]: <cr>
Primary bootname? [/unix]: <cr>
Secondary bootname? [ ]: <cr>
Number of cylinders? [1024]: <cr>
Number of heads on removable? [0]: <cr>
Number of heads on fixed? [16]: <cr>
Data bytes per sector? [512]: <cr>
Sectors per track? [34]: <cr>
Number of alternate cylinders? [20]: <cr>
File system blocksize? [1024]: <cr>
Nodename? [ ]: <6 character system node name cr>
Change the default unix device mapping? [y/n]: y <cr>
Rootdev? [0x0]: 1 <cr>
Pipedev? [0x0]: 1 <cr>
Dumpdev? [0x0]: 1 <cr>
Swapdev? [0x0]: 1 <cr>
Swplo? [0]: 68000 <cr>
Nswap? [0]: 32000 <cr>
Change the file system disk configuration? [y/n]: y <cr>
File system logical configuration: [sector start,sector count]
dsk/?s0 [ 0, 0]: 0,~ <cr>
dsk/?s1 [ 0, 0]: 0,100000 <cr>
dsk/?s2 [ 0, 0]: <cr> *
dsk/?s3 [ 0, 0]: <cr>
dsk/?s4 [ 0, 0]: <cr>
dsk/?s5 [ 0, 0]: <cr>
dsk/?s6 [ 0, 0]: <cr>
dsk/?s7 [ 0, 0]: <cr>
dsk/?s8 [ 0, 0]: <cr>
dsk/?s9 [ 0, 0]: <cr>
dsk/?s10 [ 0, 0]: <cr>
dsk/?s11 [ 0, 0]: <cr>
dsk/?s12 [ 0, 0]: <cr>
dsk/?s13 [ 0, 0]: <cr>
dsk/?s14 [ 0, 0]: <cr>
dsk/?s15 [ 0, 0]: <cr>
Is the above information correct? [y/n]: y <cr>
Are you sure you want to rewrite block 0? [y/n]: y <cr>

```

* At `dsk/?s2` you can type a "q" to quit from this list.

3.3.4 dconfig Example With Alternate File System As Root

This example uses `/dev/dsk/0s2` as the root file system, and `/dev/dsk/0s5` as the swap space. `/dev/dsk/0s1` is a file system that starts at sector 0 and is 20Mb long.

At the boot prompt, enter:

dconfig

Information about your logical file system(s) and disk sizes will display and the program will continue:

```

$$ dconfig /dev/dsk/0s0 <cr>
Disk id? [xy]: <cr>
Primary bootname? [/unix]: xy(0,40000)/unix <cr>
Secondary bootname? [ ]: <cr>
Disk type: 0
Number of cylinders? [760]: <cr>
Number of heads on fixed? [19]: <cr>
Sectors per track not including spares? [58]: <cr>
Number of spare sectors? [2]: <cr>
Nodename? [ ]: <6 character system node name cr>
Change the default unix device mapping? [y/n]: y <cr>
Rootdev? [0x0]: 2 <cr>
Pipedev? [0x0]: 2 <cr>
Dumpdev? [0x0]: 5 <cr>
Swapdev? [0x0]: 5 <cr>
Swplo? [0]: <cr> <cr>
Nswap? [0]: 32000 <cr>
Change the file system disk configuration? [y/n]: y <cr>
File system logical configuration: [sector start,sector count]
    dsk/?s0    [    0,          0]: 0,~ <cr>
    dsk/?s1    [    0,          0]: 0,40000 <cr>
    dsk/?s2    [    0,          0]: 40000,68000 <cr>
    dsk/?s3    [    0,          0]: 108000,20000 <cr>
    dsk/?s4    [    0,          0]: 128000,20000 <cr>
    dsk/?s5    [    0,          0]: 148000,32000 <cr>
    dsk/?s6    [    0,          0]: <cr> *
    dsk/?s7    [    0,          0]: <cr>
    dsk/?s8    [    0,          0]: <cr>
    dsk/?s9    [    0,          0]: <cr>
    dsk/?s10   [    0,          0]: <cr>
    dsk/?s11   [    0,          0]: <cr>
    dsk/?s12   [    0,          0]: <cr>
    dsk/?s13   [    0,          0]: <cr>
    dsk/?s14   [    0,          0]: <cr>
    dsk/?s15   [    0,          0]: <cr>
Is the above information correct? [y/n]: y <cr>
Are you sure you want to rewrite block 0? [y/n]: y <cr>

```

* At `dsk/?s6` you can type a "q" to quit from this list.

3.3.5 Reorganize Current Configuration

This section describes how to reorganize your disk if the root file system or swap device is not large enough. To check the size of your root file system or swap device, run standalone **dconfig**, using the following procedure:

1. Insert the release tape into the tape drive and at the the boot prompt enter:

```
dconfig
```

The following will display:

```
$$ dconfig.
```

2. Enter the name of your root file system disk:

```
/dev/dsk/Xs0
```

Where: X is the physical drive number for the disk (usually 0)

3. Press the carriage return repeatedly until the following prompt displays, then respond with a y:

```
Change the default unix device mapping? [y/n]
```

4. Press the carriage return and record the values for the entries (as shown in the example below) that will display:

```
Rootdev? [0x1]: <cr>
Pipedev? [0x1]: <cr>
Dumpdev? [0x2]: <cr>
Swapdev? [0x2]: <cr>
Swplo? [0]: <cr>
Nswap? [12000]: <cr>
```

5. At the following prompt, respond with a y:

```
Change the file system disk configuration? [y/n]
```

6. Press the carriage return and record the values for the entries (as shown in the example below) that will display:

```
dsk/?s0 [0,400000]: <cr>
dks/?s1 [0,60000]: <cr>
dsk/?s2 [60000,12000]: <cr>
dsk/?s3 [72000,48000]: <cr>
dsk/?s4 [120000,160000]: <cr>
dsk/?s5 [280000,120000] ,cr>
dsk/?s6 [0,0]: <cr>
dsk/?s7 [0,0]: <cr>
dsk/?s8 [0,0]: <cr>
dsk/?s9 [0,0]: <cr>
dsk/?s10 [0,0]: <cr>
dsk/?s11 [0,0]: <cr>
dsk/?s12 [0,0]: <cr>
dsk/?s13 [0,0]: <cr>
dsk/?s14 [0,0]: <cr>
dsk/?s15 [0,0]: <cr>
```

7. Press the key.

Now you can determine if the root file system is large enough. Since the sector size of a file system is in 512 byte sectors, 34Mb is equivalent to 68000 sectors. The **rootdev** on the previous example is ?s1. As you can see, it is only 60000 sectors (30Mb), so it must be enlarged. The swap device needs to be enlarged also. The following values could be entered, using **dconfig** again.

```
Nswap [12000]: 32000
dsk/?s0 [0,400000]: <cr>
dsk/?s1 [0,60000]: 0,68000 <cr>
dsk/?s2 [60000,12000]: 68000,32000 <cr>
dsk/?s3 [72000,48000]: 100000,48000 <cr>
dsk/?s4 [120000,160000]: 148000,132000 <cr>
dsk/?s5 [280000,120000]: <cr>
dsk/?s6 [0,0] <cr>
dsk/?s7 [0,0] <cr>
dsk/?s8 [0,0] <cr>
dsk/?s9 [0,0] <cr>
dsk/?s10 [0,0] <cr>
dsk/?s11 [0,0] <cr>
dsk/?s12 [0,0] <cr>
dsk/?s13 [0,0] <cr>
dsk/?s14 [0,0] <cr>
dsk/?s15 [0,0] <cr>
```

Is the above information correct? [y/n]: y

Are you sure you want to rewrite block 0? [y/n]: y

NOTE

If the values that displayed for your root file system and swap device were large enough, no reorganization is needed. Continue on to the next section.

The file system that originally looked like Table 3-1, is now organized to look like Table 3-2.

Table 3-1. File System to Reorganize

sector 0	/	30Mb
	/dev/dsk/0s1	
60000	swap space	6Mb
72000		
	/u	24Mb
	/dev/dsk/0s3	
120000		
	/appl	80Mb
	/dev/dsk/0s4	
280000		
	/user	60Mb
	/dev/dsk/0s5	
400000		

Table 3-2. Reorganized File System

sector 0	/	34Mb
	/dev/dsk/0s1	
68000	swap space	16Mb
100000		
	/u	24Mb
	/dev/dsk/0s3	
148000		
	/appl	66Mb
	/dev/dsk/0s4	
280000		
	/user	60Mb
	/dev/dsk/0s5	
400000		

Notice that in Table 3-2 **/dev/dsk/0s3 (/u)** and **/dev/dsk/0s4 (/appl)**, are in different sectors of the disk than they were in Table 3-1. When you run **mkfs** they will be restored from the **cpio** backups made in the previous section, *Back Up The File Systems*. This procedure will be defined shortly.

3.4 Create The Root File System

The minimum root file system is found in file 20 of the release tape. In order to install it, a file system must be created for it. This is done with the command **mkfs**.

3.4.1 Run mkfs

To run **mkfs**, insert the release tape into the drive and at the boot prompt enter:

```
mkfs
```

The program will load and return the prompt:

```
$$ mkfs
```

mkfs has four parameters. The third and fourth are optional, but if you specify the third, you **must** also specify the fourth. The four **mkfs** parameters are as follows:

- The first is the file system to make (for example, **/dev/dsk/0s1** if you are using this as your root device).
- The second is the size of the file system in blocks, which is the file system sector count divided by 2, unless you are using a single logical file system for your root and your swap space, in which case the value to use is **swplo** divided by 2.
- The third (optional) is the gap, which is 1 for IMSP controlled disks (**pd** type), 7 for AMSP controlled disks (**sm** type) and EMSP controlled disks (**xy** type), and 9 for MMSP controlled disks (**sd** type—P/90). The only exception is the 14" 570Mb disk, controlled by the EMSP, which uses a gap of 12.
- The fourth parameter (optional) is the blocks per cylinder to use, which is calculated using the formula:

$$\text{Blocks per Cylinder} = (\text{Sectors per Track} / 2) * \text{Number of Heads}$$

The number of sectors per track varies for each disk type for all the disk drives.

3.4.1.1 mkfs Example

Following is an example for using **mkfs** with an IMSP controlled disk, with the first logical file system as root. User input is shown in bold.

```
: mkfs
$$ mkfs /dev/dsk/0s1 34000 1 272
Mkfs: 0s1?
(DEL if wrong)
bytes per logical block = #####
total logical blocks = #####
gap (physical blocks) = 1
cylinder size (physical blocks) = 272
```

Table 3-3. Gap and Blocks per Cylinder

Drive Type	Drive Size	Controller	Blocks per Cylinder	Gap
142Mb	8"	AMSP	124	7
145Mb	8"	AMSP	165	7
286Mb	8"	AMSP	335	7
443Mb	9"	AMSP	560	7
71Mb	8"	EMSP	112	7
72Mb	14"	EMSP	82.5	7
142Mb	8"	EMSP	128	7
145Mb	8"	EMSP	160	7
145Mb	14"	EMSP	165	7
285Mb	14"	EMSP	264	7
286Mb	8"	EMSP	168	7
443Mb	9"	EMSP	551	7
570Mb	14"	EMSP	528	12
586Mb	10.5"	EMSP	660	7
22Mb	8"	IMSP	51	1
36Mb	8"	IMSP	85	1
71Mb	8"	IMSP	119	1
72Mb	14"	IMSP	85	1
142Mb	8"	IMSP	136	1
145Mb	8"	IMSP	160	1
145Mb	14"	IMSP	170	1
270Mb	8"	IMSP	198	1
285Mb	14"	IMSP	272	1
67Mb	5 1/4"	MMSP	87	9
135Mb	5 1/4"	MMSP	175	9
300Mb	5 1/4"	MMSP	210	9
310Mb	5 1/4"	MMSP	262	9

3.4.2 Run fsck

fsck ensures that commands that write to the disk have been successful. **fsck** reads through the file system and checks the blocks in use, the file system structures, and the free list. It lists out any errors, attempts to assist in repairing any problems, and prints file system statistics.

To run **fsck**, insert the release tape into the tape drive and at the boot prompt enter:

```
fsck
```

After the program loads, you will be prompted with:

```
$$ fsck
```

Enter the name of the disk you wish to check in the form:

```
/dev/dsk/Xsx
```

Where: *X* = the major disk number,
x = the minor number.

If **fsck** finds problems, it will ask you what you want done. The usual choices are "RECONNECT?" and "CLEAR?". Generally, if you encounter any errors from **fsck** while installing new software, and you are sure you have been following the instructions, you should suspect problems with your disk hardware. The one exception is that of an unlinked file of zero length, which **fsck** will automatically remove.

This rule does not apply if you encounter errors from **fsck** after your system has become active.

3.4.2.1 fsck Example

Following is an example for using **fsck** in standalone mode with this release tape. The first logical file system is used as root.

```

: fsck
$$ fsck /dev/dsk/0s1

/dev/dsk/0s1
  File System:  Volume:

  ** Phase 1 - Check Blocks and Sizes
  ** Phase 2 - Check Pathnames
  ** Phase 3 - Check Connectivity
  ** Phase 4 - Check Reference Counts
  ** Phase 5 - Check Free List
  1908 files 21054 blocks 12243 free

```

Do not be concerned that the blocks in use and the blocks free do not add up to the total number of blocks you created in your file system. The extra blocks are used for inodes, which contain file header information.

3.5 Install The Minimum Root File System

All of the files that can be installed on the disk are contained in files 20 - 24 on the release tape. In order to allow the use of UNIX utilities, file 20 contains the minimum root file system which allows you to read from tape and create directory structures. This procedure can also be used to restore a root file system that has been backed up using the **dump** command. Use the standalone program **restor** to install the root file system.

The options to the **restor** program are the same as those for the **restor(1)** command, with one exception. In addition to allowing you to specify a tape drive to use, you may also specify which file on the tape to use. This is provided since the minimum file system release is contained on file 20 of the release tape (see Table 2-1). If you have specified a number of files to skip, a message will appear and it will space forward. When it has completed, or if no number was specified, it will now tell you which file system it will **restor** to, and prompt you for a carriage return before continuing. Then the lengthy process of reading the files from the tape, building a file system, and building the free list is done. **Do not** interrupt the program before it has finished. If you do, your file system will be corrupted and you will have to run both **mkfs** and **restor** again. Examples of this program are given below. User input is shown in bold.

3.5.1 Run restor

To install the minimum root file system, insert the release tape into the tape drive, and at the boot prompt enter:

```
restor
```

After the program loads you will be prompted with:

```
$$ restor
```

Enter:

```
rf /dev/xxx/0m /dev/dsk/0s1 +20
```

Where: *xxx* is **rpt** for cartridge drives
xxx is **rrm** for reel-to-reel drives

The disk being restored to is optional, but it is usually **0s1**, as shown here. When **restor** has finished, run an **fsck** on the newly-loaded file system. Refer to the subsection *Run fsck in Standalone Mode*, earlier in this chapter, for instructions.

3.5.1.1 restor Example

Following is an example for using **restor** with this release tape. The first logical file system is being used as root and you are loading from a reel-to-reel tape drive.

```
: restor <cr>
$$ restor rf /dev/rrm/0m /dev/dsk/0s1 +20 <cr> *
Spacing forward 20 files on tape.
Last chance before scribbling on /dev/dsk/0s1. <cr>
Please wait - building free lists -
EXIT 0
```

* Use **+20** only when installing from the release tape. Omit it when installing from a backup tape.

Following is an example of loading the minimum root file system from a backup tape. An alternate logical file system is used as root, and you are loading from a cartridge tape drive.

```
: restor <cr>
$$ restor rf /dev/rpt/0m /dev/dsk/0s2 <cr>
Last chance before scribbling on /dev/dsk/0s2. <cr>
Please wait -building free lists -
EXIT 0
```

After the file system has been installed the boot prompt will return and you can run standalone **fsck** on the newly-loaded file system.

3.6 Boot Into Single-User Mode

You should now be able to boot up into single-user mode UNIX. Type a carriage return at the boot prompt and it will boot from the file name specified at the **primary bootname?** prompt of **dconfig**. A series of system registers will display, and then the root shell prompt (**#**) will display.

3.7 Install The Standard Release Files

The next part, the programs, data files and scripts, are in the 21st file on tape, and are in **cpio** archive format.

- This installation must be done in single-user mode.
- 1. Make sure that the date on your machine is correct.
- 2. Insert the release tape into the tape drive. If you have a 20Mb Sys5 cartridge tape system, insert the second volume of the tapes.
- 3. Advance the tape to the proper file and install the software with **cpio**, as follows:

- If you have a reel-to-reel tape, enter:

```
tape srcheof 21
cpio -idumBv < /dev/rrm/0m >/tmp/cpio.out 2>/tmp/cpio.err
sync
```

- If you have a cartridge tape, enter:

```
tape -f /dev/rpt/0mn srcheof 21
cpio -idumBv < /dev/rpt/0m >/tmp/cpio.out 2>/tmp/cpio.err
sync
```

As each file loads, **cpio** will write each file name to the file **/tmp/cpio.out**. A few "errors" will be listed in **/tmp/cpio.err**, but they are not necessarily a sign of failure. The acceptable errors are that **cpio** will not be able to write **/bin/cpio**, **/bin/sh** and **/etc/init**, nor will it be able to create the special device nodes that were loaded during the **restor**.

- 4. To install these files, enter:

```
/tmp/release/fixup
sync
```

3.8 Install The Manual Pages (optional)

To install the online manuals, you will need 5 Mb of free disk space. The manuals should be installed in the directory **/usr/catman**, as the program that displays them looks in this directory.

- 1. Make sure that the date on your machine is correct.
- 2. Insert the release tape into the tape drive and advance the tape to the proper file. Install the manuals as follows:

- If you have a reel-to-reel tape drive, enter:

```
cd /usr/catman
tape srcheof 22
cpio -idumBv < /dev/rrm/0m
sync
```

- If you have a cartridge tape drive, enter:

```
cd /usr/catman
tape -f /dev/rpt/0mn srcheof 22
cpio -idumBv < /dev/rpt/0m
sync
```

3.9 Retrieve Files From The Root File System Backup

In the process of installing the release a new root file system was created. If you are upgrading from a previous release, any files that were manually put onto your root file system will no longer be there. They are on the root file system **dump** backup that was created earlier. The following instructions will help you to reload these files.

1. Put the **dump** tape of the root file system into the tape drive.
2. To print the names of the files on the **dump** tape, enter:

```
dumpdir f /dev/xxx/0m
```

Where: *xxx* is the name of the tape device

3. To restore a specific file from the tape, enter:

```
restor xf /dev/xxx/0m filename
```

Where: *xxx* is the name of the tape device

filename is the exact file name as printed using **dumpdir**

The extracted file is placed in a file with a numeric name supplied to it by **restor**. **restor** will announce whether or not it found the file and display the numeric name that it assigned to the file. You can then move the file to wherever you want.

restor(1M) is explained in better detail in the *Sys5 UNIX Administrator's Reference Manual*.

3.10 Install The Update File

For this S/M/K1.7 release, and for all subsequent releases of S/M/K1.7, files 0-23 will not change. File 24 on the tape will contain the accumulation of all files that have changed since the previous release. File 24 for S/M/K1.7 will contain those changes made since the beta release. Since these instructions will vary with each update, the instructions are contained in a separate "Update Notice". Please refer to that notice and install file 24 at this time.

When you have completed this procedure, the standard release files have been installed. You must now continue on to the next chapter to manually integrate those files that must be tailored to your configuration.

If you did not reorganize your disk, proceed to the next chapter. If you had to reorganize your disk, proceed to the next section.

3.11 Reload Backups

If you had to reorganize your disk, you will need to reload those file systems that either changed sizes, or changed positions on the disk. In the examples shown in Tables 3-1 and 3-2, in the subsection *Reorganize Current Configuration*, both `/dev/dsk/0s3 (/u)` and `/dev/dsk/0s4 (/appl)` must be reloaded. You must first run `mkfs` on the logical file systems. The `mkfs` command is of the form:

```
mkfs xxxx yyyy i g
```

Where: `xxxx` is the file system logical device name
`yyyy` is the file system block size
`i` and `g` are the gap and blocks per cylinder.

For example, to use `mkfs` to create file systems for a 145Mb, 8" AMSP controlled disk, at the single-user prompt (`#`), enter:

```
mkfs /dev/dsk/0s3 24000 7 165
mkfs /dev/dsk/0s4 66000 7 165
sync
```

Now you can load the `cpio` backups, using the following procedure:

1. Make sure that the date is correct.
2. Make a directory name for each file system. For the previous example, you would make directories as follows:

```
mkdir /u
mkdir /appl
```

3. Mount the file systems, as follows:

```
mount /dev/dsk/0s3 /u
mount /dev/dsk/0s4 /appl
```

4. For each file system, in turn, do the following:

- A. Change into the appropriate directory.
- B. Insert the first volume of the backup tape for that file system into the tape drive and run `cpio`, as follows:

```
cpio -idumBv < xxxx
sync
```

Where: `xxxx` is the name of the tape drive.

If your `cpio` backup is contained on more than one tape, when it is at the end of a tape it will display the following message:

```
Cannot read tape, enter device name to continue
```

Remove the tape from the drive and insert the next volume.

- C. When the tape has loaded, type the name of the tape drive and continue.
5. Change to the top level directory and unmount the file systems, as follows:

```
cd /
umount /dev/dsk/0s3
umount /dev/dsk/0s4
sync
```

COMPLETE THE INSTALLATION

This chapter outlines the customization steps necessary to bring your machine to full operational capability. If you have received a new Plexus system, Sys5 S/M/K1.7 was installed by Plexus at the factory. However, there are some files that must be customized for your particular use. That process is explained here.

UNIX Sys5 S/M/K1.7 has been divided into three types of files. These are as follows:

- Files where the on-site version is presumed best (such as **/usr/lib/uucp/L.sys**).
- Files where the Plexus version is presumed best (such as **/bin/sh**)
- Files that require manual integration to achieve a "best" version.

This chapter deals primarily with instructions on performing the manual integration of the files that fall into the third category. We will further divide this category into two parts.

- The first group consists of files that are virtually always modified on-site. Following is a list of these files.

```
/etc/bcheckrc
/etc/brc
/etc/cshprofile
/etc/gettydefs
/etc/inittab
/etc/ioctl.syscon
/etc/passwd
/etc/profile
/etc/rc
/etc/TIMEZONE
/etc/shutdown
/etc/ttytype
/usr/spool/cron/crontabs/adm
/usr/spool/cron/crontabs/root
/usr/spool/cron/crontabs/sys
/usr/spool/cron/crontabs/uucp
/usr/src/uts/m68/cf/conf.c
```

- The second group consists of files that are modified to produce a modified kernel; for example, one that supports additional drivers. This group of files will not have been modified by most sites, and the instructions for integrating these files will be extremely straight-forward for most. Following is a list of the files that compose this second group.

```
/usr/include/sys/init.h
/usr/include/sys/space.h
/usr/src/uts/m68/cf/cf.mk
/usr/src/uts/m68/cf/conf.c
/usr/src/uts/m68/cf/linesw.c
/usr/src/uts/m68/cf/low.s
/usr/src/uts/m68/cf/name.c
/usr/src/uts/m68/cf/rflow.s
/usr/src/uts/m68/cf/kflow.s
```

Note that `/usr/src/uts/m68/cf/conf.c` appears in both lists since it is applicable to both groups.

4.1 Load The Files Onto Disk

NOTE

You must be in single-user mode during this procedure.

Insert the release tape into the tape drive and enter one of the following:

- If you have a reel-to-reel tape drive, enter:

```
cd /
tape srcheof 23
cpio -idumBv < /dev/rrm/0m
sync
cd /usr/plx/plx-ssc
```

- If you have a cartridge tape drive, enter:

```
cd /
tape -f /dev/rpt/0m srcheof 23
cpio -idumBv < /dev/rpt/0m
sync
cd /usr/plx/plx-ssc
```

The files will be installed into the directory `/usr/plx/plx-ssc`, thus avoiding over-writing your current versions.

4.2 Manually Integrate Modified Files

This section contains information about the files that require manual integration. Each file in this group will be discussed in turn. The general procedure is as follows:

1. Copy your current file version into the working directory (**/usr/plx/plx-ssc**) using a **.old** extension to distinguish your current file from the new file. Input would be as follows:

```
cp filename /usr/plx/plx-ssc/filename.old
```

2. Change into the working directory and run the **diff** program between your **.old** file and the new file. (See **diff(1)** in the *Sys5 UNIX User's Reference Manual*.)

```
cd /usr/plx/plx-ssc
diff filename.old filename
```

diff will produce a list showing the differences between your **.old** file and the new file. Review the lines marked with a "<" by **diff**. Note the location of the line in the file.

3. Edit the new version of the file and add your customizations.
4. Move the edited new file into the desired directory.

If you experience any problems with a file you have modified, you still have your version saved as **/usr/plx/plx-ssc/filename.old**.

4.2.1 /etc/bcheckrc

If you are upgrading from a Sys3 system, you will not have a copy of the file **/etc/bcheckrc**. To write a copy into your **/etc** directory, enter the following:

```
cp bcheckrc /etc/bcheckrc
```

See the *Sys5 Administrator's Handbook* for more information about the **/etc/bcheckrc** file.

4.2.2 /etc/brc

If you are upgrading from a Sys3 system, you will not have a copy of the file **/etc/brc**. To write a copy into your **/etc** directory, enter the following:

```
# cp brc /etc/brc
```

This file has had no modifications made at Plexus, so you will not have to make any changes to your existing copy.

4.2.3 /etc/cshprofile

Following is the listing for the file **/etc/cshprofile**. Check your file and make changes, if necessary, so that it looks like this listing.

```
#
# @(#)cshprofil.sh      5.4 10/30/86
setenv TZ 'cat /etc/TIMEZONE'
cat /etc/motd
if { mail -e } then
    echo "you have mail"
endif
if ( $LOGNAME != root ) then
    news -n
endif
```

4.2.4 /etc/gettydefs

If you are upgrading from a Sys3 system, you will not have a copy of the file **/etc/gettydefs**. To write a copy into your **/etc** directory, enter the following:

```
# cp gettydefs /etc/gettydefs
```

The only change in this file that has been added since Release 1.2 is that the default console speed has been changed from B9600 to EXTB, which means that the console baud rate will be taken from the switches set on the CPU. You may wish to make only this change to your existing **/etc/gettydefs**.

See the *Sys5 Administrator's Handbook* for a listing of this file.

4.2.5 /etc/inittab

- If you are **upgrading** to S1.7 or M1.7, follow the instructions in the next chapter in the subsection *Create /etc/inittab*.
- If you are installing any version of 1.7 **onto a new system**, you will need to customize the **/etc/inittab** file for your login configuration. Follow the instructions in the next chapter in the section *Modify /etc/inittab*.
- If you are **upgrading** to K1.7 from an earlier **K** version (K1.6, K1.6.5), skip to the section *Modify /etc/rc For K1.7* in the next chapter.

4.2.6 /etc/ioctl.syscon

If you are currently running UNIX Sys5, you will not need to alter this file. This is a data file that did not exist in UNIX Sys3, so if you are upgrading from Sys3 to Sys5, enter the following:

```
cp ioctl.syscon /etc
```

4.2.7 /etc/passwd

If you are upgrading from a release version of UNIX Sys3 older than 3.2 (such as, 3.0, 3.1, 3.12) you must add the following line to the file **/etc/passwd**:

```
lp:locked:71:2:0000-lp(0000):/usr/spool/lp:
```

Any upgrade being applied to Sys3, Version 3.2 or newer, or to Sys5, will not need to add this line, as it should already exist.

4.2.8 /etc/profile

Following is the listing for the file **/etc/profile**. Check your file and make changes, if necessary, so that it looks like this listing:

```
#
# @(#)profile.sh          5.4 10/30/86

#    @(#)profile.sh      1.4

trap "" 1 2 3
export TZ
TZ='cat /etc/TIMEZONE'
readonly LOGNAME
case "$0" in
-sh | -rsh)
    trap : 1 2 3
    cat
    trap " " 1 2 3
    if mail -e
    then echo "you have mail"
    fi
    if [ $LOGNAME != root ]
    then
        news -n
    fi
    ;;
-su)
    :
    ;;
esac
trap 1 2 3
```

4.2.9 /etc/rc

This file is larger and more complex than most of the others in this group. Plexus has made several additions and changes to this file over time. In general, if your **/etc/rc** file has worked in the past, it will still work, whether you are using a Sys3 version of UNIX or a Sys5 version.

The recommended procedure for altering this file is as follows:

1. Copy **/etc/rc** to **/usr/plx/plx-ssc/rc.old**
2. Compare your **/etc/rc** with the appropriate **/etc/rc** listing in the *Sys5 Administrator's Handbook*, making sure that you have all the entries.
3. If, when you change from state to state, you find you are experiencing problems, you can copy the file **/usr/plx/plx-ssc/rc.old** back to **/etc/rc**.

4.2.10 /etc/TIMEZONE

This file contains the correct value for setting your TZ environment variable, and all the scripts that previously used the command **TZ=AAANAAA** should now use the command **TZ='cat /etc/TIMEZONE'**.

Edit this file and change the string to the correct value for your time zone. The default is **PST8PDT**. The value consists of three parts, your "standard" time zone (in the default example, **PST**), the number of hours difference between your "standard" time zone and Greenwich Mean Time (in the default example, **8**), and your daylight savings (summer time) time zone (in the default example, **PDT**). If you do not use daylight savings time, leave this part blank.

Copy this file to **/etc**.

4.2.11 /etc/shutdown

The Sys3 version of this file differs from the Sys5 version, but either version will work acceptably.

There has been one change to this file which you should edit into your version of **shutdown**. In the line which performs a mount piped to **sed** (**mount | sed**) to extract NOS mounts and to construct rumounts, a space should be added after the third asterisk. Search the file **/etc/shutdown** for the first occurrence of "**mount | sed**" and make this change.

4.2.12 /usr/spool/cron/crontabs Files: adm, root, sys, uucp

These files contain functions to be executed by **cron** with the environment and UID of the various users indicated by the file names. They did not exist in UNIX Sys3, but the function they perform was done by the file **/usr/lib/crontab**. If you are upgrading from Sys3 to Sys5, you will need to review the file **/usr/lib/crontab** and add the various commands in it to the appropriate user's file. If you are upgrading from an earlier version of Sys5, you will need to review the contents of the existing files to be sure that the Plexus changes have been incorporated. The files that need changes are shown below.

- **/usr/spool/cron/crontabs/adm**

The lines that run the system accounting have been commented out. If you want to run this package, these lines should have the leading **#** removed.

```
#    @(#)adm      1.1
#0 4 * * 1-6 /usr/lib/acct/runacct 2> /usr/adm/acct/nite/fd2log
#0 * * * * /usr/lib/acct/ckpacct
#15 5 1 * * /usr/lib/acct/monacct
#5 18 * * 1-5 /usr/lib/sa/sa2 -s 8:00 -e 18:01 -i 3600 -ubcwydaqvm &
```

- **/usr/spool/cron/crontabs/root**

The line in bold face has been added. The line in italics should be deleted. Note that both of these lines are too long for the size of the page (and the screen), and wrap to two (the italicized line) and three (the bold line) lines.

```
#    @(#)root 1.1
0,10,20,30,40,50 * * * * /etc/eccdaemon /usr/adm/ecclog
0 5 * * 1 /etc/errstop 2>/dev/console;/bin/mv /usr/adm/errfile /usr/adm/oerrfile;/usr/lib/errdemon 2>
/dev/console; /usr/bin/errpt /usr/adm/oerrfile | /bin/mail adm
0 5 * * 1 /etc/errstop 2>/dev/console;/bin/mv /usr/adm/errfile /usr/adm/oerrfile;/bin/mv
/usr/adm/console_log /usr/adm/oconsole_log;/usr/lib/errdemon 2>
/dev/console; /usr/bin/errpt /usr/adm/oerrfile | /bin/mail adm
0 2 * * 1-6 /usr/lib/acct/dodisk
```

- **/usr/spool/cron/crontabs/sys**

No changes have been made to this file. No integration action should be needed.

- **/usr/spool/cron/crontabs/uucp**

No changes have been made to this file. No integration action should be needed.

4.2.13 /usr/src/uts/m68/cf/conf.c

This file contains kernel configuration values that are customarily "tuned" on-site. In the UNIX Sys3 version of this file, the configurable variables had hard-coded values, as they did in the early UNIX Sys5 version. These variables are now calculated as a function of the number of users the kernel is configured for. The number of users is currently set to 80 for Plexus systems P/35, P/55, P/60, P/75, P/90, and P/95. Rather than change the values set for these variables, it is recommended that you recompile your kernel as described in the section *Relink the Kernel* later in this chapter.

4.3 Manually Integrate Kernel Files

If you **have not** added drivers to your kernel, follow the instructions in the following section, *Ensure Kernel File Integrity*, to ensure your kernel integrity.

If you **have** added drivers to your kernel, or made other serious modifications, review the procedure in the section *Integrate Kernel Files With Added Drivers*, below. **Do not** follow the procedure in the following section, *Ensure Kernel File Integrity*.

4.3.1 Ensure Kernel File Integrity

If you **have not** added drivers to your kernel, ensure your kernel file integrity. Enter:

```
cp init.h space.h /usr/include/sys
cp cf.mk conf.c linesw.c low.s name.c rlow.s klow.s /usr/src/uts/m68/cf
```

Proceed to the following section, *Integrate Kernel Files With Added Drivers*.

4.3.2 Integrate Kernel Files With Added Drivers

The following kernel files were not over-written by the installation process. You **MUST** use the versions of these files provided in **/usr/plx/plx-ssc** as the base version to relink the UNIX kernel, and add any changes you have made to these versions.

```
/usr/include/sys/init.h
/usr/include/sys/space.h
/usr/src/uts/m68/cf/cf.mk
/usr/src/uts/m68/cf/conf.c
/usr/src/uts/m68/cf/linesw.c
/usr/src/uts/m68/cf/low.s
/usr/src/uts/m68/cf/name.c
/usr/src/uts/m68/cf/rlow.s
/usr/src/uts/m68/cf/klow.s
```

The recommended procedure follows. **YOU MUST** do this for every file in the above list.

1. Move your existing version of the file to **/usr/plx/plx-ssc/filename.old**.
2. Run **diff** between the 1.7 version of *filename* and your version, *filename.old*. Look only for additions in *filename.old* that have been added at your site.
3. Edit *filename* and add your site-specific changes.
4. Move *filename* to the appropriate location.

If you have previously relinked your kernel, you must do it again now to incorporate your modifications. Proceed to the following section, *Relink The Kernel*.

If you have never relinked your kernel you need not do so now. Proceed to the next chapter, *ENTER MULTI-USER MODE*.

4.4 Relink The Kernel

You need to relink your kernel and incorporate your modifications only if you have relinked it before. If you do relink your kernel you may want to adjust your SYNCRATE. This new feature and other tunable parameters are described in the chapter *CONFIGURABLE FILES*.

NOTE

Ensure that your system date is correct, because if it is older than the kernel modules, the kernel module **cf/name.c** will not compile and the kernel will not relink.

If you have a 1Mb system you may need to relink your kernel specifying a smaller number of users, (the **-u** option) rather than using the Plexus default of 80. The default kernel is sufficiently large to cause the ICP and ACP downloads to fail, reporting the error message "dnlld <filename> too big".

NOS users: Save the file **:mkuts68** from the release tape before reloading the NOS software. The **:mkuts68** file from the NOS release will not work for S/M/K1.7.

To relink your kernel, do one of the following:

- If you **do not** have a NOS system, enter:

```
cd /usr/src
sh :mkuts68 -rX1.7 -u ##
```

Where: X is **S**, **M**, or **K**, corresponding to the upper case letter on your release tape. (It will be in the form *sys5 s1.7*.)

is the number of users you wish to build a kernel for. Available choices are 8, 16, 32, 64, 80, and 128.

- If you **do** have a NOS system, follow the instructions in the NOS installation manual for reinstalling NOS and relinking the kernel.

If there were no load errors during relink, save your old **/unix** kernel to another name and rename the new kernel to **/unix**. Enter:

```
sync
mv /unix /unix.save
mv /usr/src/uts/m68/sys5.2x## /unix
sync
```

Where: x is **s**, **m**, or **k**, as described above, but is lower case here.

is the number of users you wish to build a kernel for.

Reset and reboot your system to complete the installation.

ENTER MULTI-USER MODE

This chapter describes site-dependent and other procedures that may be needed to complete the installation of Sys5 S/M/K1.7. These changes are necessary before bringing the machine into multi-user mode.

5.1 Create Or Modify /etc/inittab

If you are updating your system with this software release, or adding an ACP, you must edit the **/etc/inittab** file. See the *ACP Installation Manual* for information about adding an ACP.

If you are **upgrading** to S1.7 or M1.7, you must run **ttyconf** to create a new **/etc/inittab** file. Proceed to the following subsection, *Create /etc/inittab*.

If you are installing any version of 1.7 **onto a new system**, you must configure the **inittab** file to suit your needs. Skip to the section *Modify /etc/inittab*.

If you are **upgrading** to K1.7 from an earlier **K** version (K1.6, K1.6.5), no changes are needed in your **/etc/inittab** file. Skip to the section *Modify /etc/rc For K1.7*.

5.1.1 Create /etc/inittab

If you are upgrading to S1.7 or M1.7 you must create a new **/etc/inittab** file. Follow the instructions below.

1. Copy your existing **/etc/inittab** to a temporary file. Enter:

```
cp /etc/inittab /usr/plx/plx-ssc/inittab.old
```

2. Use the **ttyconf** command to create a new **/etc/inittab** file. Enter the appropriate number of ICPs, ACPs, and baud rates, and **ttyconf** will create the physical devices for the new **inittab** file. **ttyconf** zeroes out the current **/etc/inittab** and moves in **/etc/inittab.mstr** to replace it. **ttyconf** then generates the appropriate number of entries based on the number of ICP's and ACP's specified. In the instruction shown, all arguments must be specified. Use 0 (zero) as an argument place holder. Enter:

```
ttyconf (#ICPs) (#ACPs) (icpbaud) (acpbaud)
```

ttyconf also created the physical devices in the **/dev** directory according to the parameters specified. These devices include the following:

```
ic[0-4] ac[0-4] pp[0-4] tty[0-n]
```

Where: *n* = the total number of serial ports, minus one.

The following is an example of the **ttyconf** command for zero ICPs and two ACPs:

```
ttyconf 0 2 0 9600
```

Continue to the next section, *Modify /etc/inittab*.

5.1.2 Modify /etc/inittab

NOTE

Before proceeding with this section we strongly advise that you back up your current version of **/etc/inittab**.

- When installing any version of 1.7 onto a new system, you must modify the **inittab** file configuration to suit your login needs by enabling or disabling the appropriate **tty** ports. To do this you must edit **/etc/inittab**, and for each **tty** line that **will** be used as a login port, the third field should show the word **respawn**. For each **tty** line that will **not** be used as a login port, the third field should show the word **off**.

For example, to **disable** login on **tty7**, the following line:

```
07:2:respawn:/etc/getty /dev/tty7 9600
```

should be changed as follows:

```
07:2:off:/etc/getty /dev/tty7 9600
```

To **enable** login on **tty7**, the following line:

```
07:2:off:/etc/getty /dev/tty7 9600
```

should be changed as follows:

```
07:2:respawn:/etc/getty /dev/tty7 9600
```

The change takes effect when you change into **init 2** from **init s** (usually after rebooting) or when **init** is asked to re-examine the **/etc/inittab** by the command **telinit q**.

- If you are upgrading from Sys3 to Sys5, you must compare the third field in the lines of your old **/etc/inittab** file (saved earlier as **/usr/plx/plx-ssc/etc/inittab.old**) with those in the new **/etc/inittab** file created by **ttyconf**. Sys3 used **c** and **o** in that field, where Sys5 uses **respawn** and **off**. You must make sure that your new lines show **respawn** where there was a **c**, and **off** where there was an **o**. In the file **/usr/plx/plx-ssc/etc/inittab.old**, one of these lines may look as follows:

```
2:07:c:/etc/getty /dev/tty7 9600
```

Your new line, in Sys5 format, should look like the following:

```
07:2:respawn:/etc/getty /dev/tty7 9600
```

5.1.3 Create /dev Entries

If you have a K1.7 version, the **/dev** entries will be created when **/etc/rc** is run. Skip to the section *Modify /etc/rc For K1.7* later in this chapter.

If you are installing S/M1.7 onto a new system (and did not run **ttyconf**), use **mknod** to create **/dev** entries. The format is:

```
mknod devname c majornumber minornumber
```

Major numbers are as follows:

```
ICP = 14  
ICP ttys = 15  
ICP pps = 16
```

```
ACP = 30  
ACP ttys = 31  
ACP pps = 32
```

Minor numbers are the index numbers of the devices.

For example, if you have one ICP and wish to create **ic0**, enter:

```
mknod /dev/ic0 c 14 0
```

To create 8 **ttys**, 0 - 7, enter:

```
mknod /dev/tty0 c 15 0  
.  
.  
.  
mknod /dev/tty7 c 15 7
```

To create a device for the parallel port, enter:

```
mknod /dev/pp0 c 16 0
```

5.2 Modify /etc/rc

You must make some modifications to your `/etc/rc` file. To help you find your way through the file, refer to the appropriate listing of the `/etc/rc` files in the *Sys5 Administrator's Handbook*.

For all versions of 1.7, follow the instructions below, then proceed to the instructions for your particular version.

1. Near the very beginning of the `/etc/rc` listing, find the line:

```
# put umounts here
```

Add the file systems you want to unmount here.

2. A few lines later, find the line:

```
# put mounts here (/usr, etc.).
```

Add the file systems you want to mount here.

3. Near the end of the file, find the following lines:

```
#!/usr/lib/errrdemon
#echo errrdemon started
```

Uncomment these lines to enable **errrdemon**.

5.2.1 Modify /etc/rc For S/M1.7

For both S1.7 and M1.7, in addition to the above changes there are two more places that must be modified. They are as follows:

- Delete the following lines that cause the IMSP code to be downloaded. The download of IMSP is now taken care of by the file `/etc/inittab`.

```
if [ $TimesInCurrentState -eq 0 ] ; then
    sync
    sleep 3          # time for the above sync to happen
    /etc/dnld -dd -a 5000000 -f /usr/lib/dnld/imsc -o /dev/im0
fi
```

- If you have a new system, edit the `icp/acp` download section (near the end of the file) and uncomment the appropriate lines in the examples shown for the number of ICPs and ACPs that your system has, or enter lines as appropriate to your configuration. The maximum configuration is a total of 5 ICPs and ACPs. ICP/ACP numbering is 0 through 4. Following are some examples:

Example 1: 1 ICP and 2 ACPs

```
/etc/dnld -d -f /usr/lib/dnld/icp -o /dev/ic0 -a 4000
/etc/dnld -L -da 1 2 -f /usr/lib/dnld/acp -a 2000
#comment: 1 is starting ACP # and 2 is # of ACPs
```

Example 2: 3 ICPs and 1 ACP

```
/etc/dnld -d -f /usr/lib/dnld/icp -o /dev/ic0 -a 4000
/etc/dnld -d -f /usr/lib/dnld/icp -o /dev/ic1 -a 4000
/etc/dnld -d -f /usr/lib/dnld/icp -o /dev/ic2 -a 4000
/etc/dnld -L -da 3 1 -f /usr/lib/dnld/acp -a 2000
#comment: 3 is starting ACP # and 1 is # of ACPs
```

Example 3: 0 ICPs and 4 ACPs

```
/etc/dnld -L -da 0 4 -f /usr/lib/dnld/acp -a 2000
#comment: 0 is starting ACP # and 4 is # of ACPs
```

Example 4: 2 ICPs and 0 ACPs

```
/etc/dnld -d -f /usr/lib/dnld/icp -o /dev/ic0 -a 4000
/etc/dnld -d -f /usr/lib/dnld/icp -o /dev/ic1 -a 4000
```

5.2.2 Modify /etc/rc For K1.7

The additional changes that you must make to your **/etc/rc** file for the K1.7 version are as follows:

- Immediately following the section of the listing where you enabled **errdemon**, is the area referring to VCPs. Setting up your starting VCP number and the number of VCPs is done by simply uncommenting the appropriate lines of the **/etc/rc** listing. Uncomment the line that shows the same number of VCPs that you have on your system. The line will look similar to the following:

```
# /etc/dataio -dv ? ? -f /usr/lib/dnld/vcp5 -a 2000
```

The first question mark indicates the number of the starting VCP, which should always be zero. The second indicates the total number of VCPs. Enter these values here.

- A few lines later, find the following lines and uncomment either or both of them, as suits your configuration. If you have serial ports on any VCP, uncomment the **ASYNC** line. If you have parallel ports on any VCP, uncomment the **PARA** line.

```
# /etc/vconfig/vconfig -cl /etc/vconfig/ASYNC
# /etc/vconfig/vconfig -cl /etc/vconfig/PARA
```

5.3 Configure A VCP

If your system does not use VCP configuration, skip this section and proceed to the section *Verify /dev/swap*.

The contents of the file **/etc/vconfig/ASYNC** will depend on the number of VCP cards in your machine and the number of link cards on each VCP. The ASYNC file shipped with this release is for one VCP with four link cards. Each link card contains eight ports.

The file **/etc/vconfig/PARA** is used to configure parallel ports on the VCP. The PARA file shipped with this release is configured such that one 4-port parallel link card is on the VCP0.

The section *Configuring a VCP* in the *Sys5 Administrator's Handbook* describes ASYNC and PARA in more detail.

5.4 Verify /dev/swap

/dev/swap must agree with the logical device number that you place the swap space on. The default logical device file is **/dev/dsk/0s2**. Ensure that **/dev/swap** accesses the device you have selected as **swapdev** with **dconfig**. You can determine the value used for **/dev/swap** with the command:

```
ls -l /dev/swap
```

The system response should look like the following:

```
br--r----- 1 sys      sys      0, 2 Mar 11 1985 /dev/swap
```

Where: **0** is the physical disk number of the logical file system,
2 is the minor number.

If the minor number is not the same as the value used for **swapdev** with **dconfig**, you must recreate **/dev/swap** as follows:

```
rm /dev/swap
mknod /dev/swap b 0 X
```

Where: **0** is the physical disk number of the logical file system,
X is the minor number.

/dev/swap will point to the same physical disk location as **/dev/dsk/0sX**.

5.5 File Systems Created Prior To Sys5 Version 1.4

In versions of Sys5 prior to 1.4, **mkfs** had a bug that set a variable in the file system super block to indicate the number of physical blocks per logical block on the disk inversely to the way **ncheck**, **clri**, and possibly other programs, interpret it. If you have file systems created prior to Version 1.4, and if you have programs that execute the **ncheck** and **clri** commands, for performance reasons it would be best to back them up, rebuild them with **mkfs**, then restore them.

If this is very inconvenient, you may run **fsdb** for every file system created with **mkfs** prior to Sys5 Version 1.4, as follows:

1. Load the release tape into the tape drive.
2. At the boot prompt, enter:

```
fsdb
```

3. At the `$$ fsdb` display, enter the logical file system name.

- For Sys5, enter:

```
/dev/dsk/Xsxx
```

Where: **X** is the major disk number,
xx is the minor number (such as 0s1, 1s12, 2s10, etc.).

- For Sys3, enter:

```
/dev/dky
```

Where: **y** is the disk index number.

The system displays the following:

```
FSIZE = 34000 ISIZE = 7934
```

4. Next, enter:

02766 = 1 (the leading 0 must be entered)

The system displays the following message:

002766:000001 (1)

To return to the boot prompt, enter a **q**. At the boot prompt, press **RETURN** and **/unix** will boot.

5.6 Boot UNIX From Tape

Sometimes the operating system on the disk can become damaged in such a way that the system cannot be booted from disk. A method to recover from this situation is to boot the system from the master tape that is included in this release. To do so:

- Insert the release tape into the tape drive.
- Depending upon which version of the PROM devices exists on your CPU board, at the boot prompt enter either **unix** (for Sys5) or **sys3** (for Sys3) to boot the operating system.

After booting the system from tape it is possible to selectively **restor** the file system from your latest backup, as described in the section *Retrieve Files From The Root File System Backup*, in the earlier chapter *LOAD S/M/K1.7 ONTO AN EXISTING SYSTEM*, and in the user's manual that came with your system or the *Sys5 Administrator's Handbook*.

5.7 Carrier Detect On ACP

The ACP will not allow port access when the port's carrier detect signal is low and the port's **clocal** characteristic indicates that it is a modem port. To access ports having these characteristics you must do the following:

- Use **cu** with the **-m** option.
- From a "C" program:
 1. Open the port with **O_NDELAY** set.
 2. Set **CLOCAL** on with an **ioctl** call.
 3. Close the port.

Both of these techniques make the port a local port that will allow communication with the modem before an external connection is established.

5.8 Back Up The Root File System

When all of the files on the root file system have been modified for your configuration you should back up your root file system with the **dump** command.

Now you are ready to enter multi-user mode. At the single-user prompt (**#**), enter:

init 2

Some status messages will display and then you will see your login prompt.

GENERAL INFORMATION

This chapter contains general information about *Plexus Sys5 UNIX*.

6.1 Special Device Names

The following subsections list the names of special devices that Sys5 uses:

/dev/dsk/[0s0-15s15]	Disk, blocked I/O
/dev/rdisk/[0s0-15s15]	Disk, character I/O
/dev/rmt/0m	Tape, unblocked I/O Magnetic media, unblocked I/O, cartridge or reel-to-reel, this is created by the user using mknod
/dev/rmt/0mn	Magnetic media, unblocked, no rewind, cartridge or reel-to-reel, mknod must be done in order to use this device
/dev/rpt/0m	Magnetic media, unblocked I/O, cartridge only
/dev/rpt/0mn	Magnetic media, unblocked I/O, no rewind, cartridge only
/dev/rpt/0fm	Magnetic media, character I/O, bit flipping, cartridge only
/dev/rpt/0fmn	Magnetic media, character I/O, bit flipping, no rewind, cartridge only
/dev/rft/0m	Magnetic media, unblocked I/O, disk-caching cartridge, IMSP disk only
/dev/rft/0mn	Magnetic media, unblocked I/O, no rewind, disk-caching cartridge, IMSP disk only
/dev/rrm/[0-3]m	Magnetic media, unblocked I/O, reel-to-reel only
/dev/rrm/[0-3]mn	Magnetic media, unblocked I/O, no rewind, reel-to-reel only
/dev/rrm/[0-3]bm	Magnetic media, character I/O, byte-swapping, reel-to-reel only
/dev/rrm/[0-3]bmn	Magnetic media, character I/O, byte-swapping, no rewind, reel-to-reel only
/dev/crm/[0-3]m	Magnetic media, unblocked I/O, memory-caching, reel-to-reel only
/dev/crm/[0-3]mn	Magnetic media, unblocked I/O, no rewind, memory caching, reel-to-reel only
/dev/rfp/0s0	Magnetic media, unblocked I/O, floppy only
/dev/tty[0-39]	Serial ports
/dev/lp	Line printer port
/dev/console	System console
/dev/swap	Swap device, used by ps(1) and a few other commands
/dev/mem	Memory
/dev/kmem	Kernel data space within memory
/dev/liomem	Special I/O ports on processor board
/dev/mbmemb	System's multibus memory address space

/dev/mbiomem	System's multibus I/O address space
/dev/null	Bit bucket
/dev/error	Provides access to error records in the kernel
/dev/prf	Provides access to activity information in the kernel
/dev/vpm[0-4]	Provides a software construct for implementation link protocols on the ICP
/dev/modem	Provides access to optional built-in modem (P/75, P/90, and P/95 only)
/dev/ccb	Provides access to common circuits board (P/75, P/90, and P/95 only)
/dev/conslog	Provides access to the messages and errors reported to the console as kept in the kernel.

6.2 Sys5/Sys3 Device Names

Table 6-1 lists cross references between Sys5 and Sys3 device names. The format is:

device name (major number/minor number)

Thus, the entry for */dev/dsk/0s0* has a major number of 0 and a minor number of 0, and the entry for */dev/rrm/0m* has a major number of 8 and a minor number of 0.

Table 6-1. Cross References Between Sys5 and Sys3

Sys5	Sys3
DISKS	DISKS
Block Devices /dev/dsk/0s0 (0/0) /dev/dsk/0s1 (0/1) /dev/dsk/1s0 (0/16) /dev/dsk/2s0 (0/32)	/dev/dk0 (0/0) /dev/dk1 (0/1) /dev/dk16 (0/16) /dev/dk32 (0/32)
Character Devices /dev/rdisk/0s0 (3/0) /dev/rdisk/1s0 (3/16)	/dev/rdk0 (3/0) /dev/rdk16 (3/16)
TAPES	TAPES
Generic /dev/rmt/0m /dev/rmt/0mn	/dev/rmt0 (4/0) /dev/nrmt0 (4/4)
9 Track /dev/rrm/0m (8/0) /dev/rrm/0mn (8/4) /dev/rrm/0hm (8/8) /dev/rrm/0hmn (8/12)	/dev/rrm0 (8/0) /dev/nrrm0 (8/4) /dev/nrrm0 (8/8) /dev/nrrm0 (8/12)
Cartridge /dev/rpt/0m (6/0) /dev/rpt/0mn (6/4)	/dev/rpt0 (6/0) /dev/nrpt0 (6/4)

6.3 Some Modifications To System V

The following programs are additions to Sys5 that are not part of AT&T System V.

/etc/dump	/stand/restor	/usr/plx/more
/etc/restor	/usr/lib/dnld/acp (S/M)	/usr/plx/odconf
/etc/sys	/usr/lib/dnld/imsc (S/M)	/usr/plx/odcreate
/etc/acpconf (S/M)	/usr/lib/dnld/vcp5 (5)	/usr/plx/odctl
/etc/acpdmp (S/M)	/usr/plx/cdconf	/usr/plx/oddf
/etc/vconfig/brdrst (K)	/usr/plx/clear	/usr/plx/oddiag
/etc/vconfig/vconfig (K)	/usr/plx/copytape	/usr/plx/odls
/etc/vcpdmp (K)	/usr/plx/csh	/usr/plx/odrepair
/stand/badtrack (S/M)	/usr/plx/ctags	/usr/plx/odstat
/stand/cat	/usr/plx/dataio	/usr/plx/printenv
/stand/dconfig	/usr/plx/dial	/usr/plx/ramdisk
/stand/dd	/usr/plx/diction	/usr/plx/sadnld
/stand/dformat	/usr/plx/dprog	/usr/plx/script
/stand/du	/usr/plx/dumpdir	/usr/plx/strings
/stand/fbackup (S/M)	/usr/plx/explain	/usr/plx/style
/stand/fsck	/usr/plx/head	/usr/plx/tape
/stand/fsdb	/usr/plx/las	/usr/plx/tset
/stand/help	/usr/plx/mas	/usr/plx/uuencode
/stand/ls	/usr/plx/mconv	/usr/plx/uudecode
/stand/mkfs	/usr/plx/mdial	/usr/plx/xstr
/stand/od	/usr/plx/mkstr	

The following routine is an addition to Sys5 that is not part of AT&T System V.

lockf(2)

6.3.1 Not Provided – Not Applicable

The following list defines programs, libraries, and other software that are not provided because they are

- specific to non-Plexus hardware
- replaced by equivalent software.

The following associated manual pages are also not provided:

/bin/kas	/stand/boot2	/usr/lib/lib2A.a
/bin/kasb	/stand/iltl	/usr/lib/lib2B.a
/bin/kun	/stand/rf11boot	/usr/man/man1/kas.1
/bin/kunb	/stand/rk11boot	/usr/mdec
/etc/fscv	/stand/rl11boot	/usr/mdec/copy
/etc/stcntrl	/stand/rp03boot	/usr/mdec/dldr
/etc/stload	/stand/rp04boot	/usr/mdec/iltl
/etc/stproto	/stand/rs04boot	/usr/mdec/list
/etc/vlx	/unixhpht	/usr/mdec/mboot
/lib/as2	/unixhptm	/usr/mdec/rf11booti
/lib/c0	/unixrktm	/usr/mdec/rk11boot
/lib/c1	/unixrlht	/usr/mdec/rkf
/lib/fc0	/unixrltm	/usr/mdec/rp03boot
/lib/fc1	/unixrpht	/usr/mdec/rp04boot
/lib/fcrt0.o	/unixrptm	/usr/mdec/rs04boot
/lib/fmcr0.o	/usr/bin/sdb	/usr/mdec/tapeboot
/stand/boot1	/usr/include/sys.s	/usr/mdec/tboot
/bin/sdb	/etc/sysdef	/etc/config
all graphics programs		

6.3.2 Not Provided – No Source

The following commands are documented in *System V* but were omitted from the *System V* source distributed by AT&T to Plexus:

```
/bin/primes
/bin/vcat
/bin/vpr
```

6.4 Known Problems

Following is a list of known problems in version S/M/K1.7 of the UNIX Sys5 operating system:

/bin/adb	adb does not treat numbers input with a leading zero as octal, it does not currently disassemble the dbcc instruction, and it does not print out negative relative addresses correctly.
/bin/ld	The -m option lists included modules only, rather than modules and files.
/bin/make	The -k option (abandon work on the current entry, but continue on other branches not depending on that entry) does not work.
/dev/ccb	This device is not supported on the P/35, P/55 and P/60 systems. If you have one of these Plexus computers, at system boot the error message "sys: can't open /dev/ccb errno=2" will display. This does not indicate a problem.

/etc/mkfs	mkfs will not print an error message if it is invoked with a block count larger than the disk configuration allows. fsck will catch the problem, so do be sure to run fsck after every mkfs , and check the values reported by it.
/usr/bin/lint	Specifying a register storage class on an argument causes lint to incorrectly report errors.
/usr/bin/nroff	nroff places only one word on the last output line when a [wh macro] diversion is used to detect the end of the page.
/usr/lib/uucp/uuxqt	uuxqt will not correctly restart communications pending for the uux requestor, leaving the X. file in the spool directory.
/usr/plx/csh	This shell is an older version and does not support a lot of features available in newer versions, such as redirection from stdin to a variable. An attempt to expand $\$[variable]$ will include the right bracket in the expanded value.
/usr/bin/ex	If you attempt to use ex or vi on a text file that has a pattern <i>ei:string:</i> or <i>ex:string:</i> in the first five or last five lines, vi will attempt to interpret the string as an editor command. Thus, a file with the string <i>ex:q:</i> can never be edited with vi , as the <i>q</i> causes vi to quit. Use ed instead.
/etc/ff	This utility does not work. Use find instead.
/usr/plx/more	More sets echo on and does not restore the previous value of echo on exit.
/usr/plx/tape	The 1.7 version of this command will not work with any version of /unix prior to Sys5 S/M/K1.7. The erasureall option fails to erase the tape on cache tape drives.
password aging	Login names of more than 7 characters cause password aging to fail. In addition, login names of other than alpha-numeric characters may cause problems with system accounting.
regex()	The regex metacharacter "\$", used to indicate end-of-string, does not work.
/bin/tail	This utility will not report the number of lines correctly after line 81.
/bin/dd	It is advised that the conv=flip option be used when writing the tape out rather than when reading the tape in. An I/O error message will occasionally occur if you do the conversion while reading in the tape. The dd conv=sync option inserts nulls when small piped blocks are read by dd which expects 20Kb blocks. The meaning of "b" on buffer size is now 512Kb rather than its previous value of 1024Kb. This conforms to the SVID. If you are currently using "b", change it to "k". This problem is mostly noticed when using reel-to-reel tapes.
/usr/bin/cu	Redials (re-executes) the dial program after successful exit from cu . This occurs when tee is executed. Redials after "" when cu is initialized under " sh tee file ".

ACP	Does not support RTS/CTS.
/usr/bin/vi	Core dumps when it encounters an unknown terminal type.
/usr/plx/vtty	Does not return bad return code to sh when aborting, or after printing.
/stand/dformat	Does not have table information for certain new disks.
/etc/termcap	Does not contain Wyse-75 entries.
CONTROL D	When executed during initial login, will not recycle getty or login .
/usr/bin/vi	Pages backwards two lines too far.
/bin/cc	Greenhills "C" 1.8.1 evaluates expressions differently from PCC and K & R definition.
/bin/dd	dd does not see an eof at the end of a logical disk.
/bin/mailx	mailx permits a text line that contains only a dot to be sent even though this will always terminate the message when piped to /bin/mail .
/usr/bin/lp	If a printer file is cancelled while the file is being printed the printer lock file is set and must be removed in order to use the printer again.
/usr/plx/tape	The eraseall option fails to release reel-to-reel tape on caching tape drive. Program hangs when invoked.
td	mm firmware doing td(0.0) for memory dumps does not put eof mark at end of file. Additional files cannot be written out to the cartridge. .LE

6.5 Problems Fixed In This Release

Following is a list of the problems fixed in S/M/K1.7:

/etc/dd	The size specification "b" is now set to 512.
/bin/mail	Creates a unique temporary file name /tmp/maXXXXXX . Previously, the temporary file name had only five X's.
/etc/mount	Mount reports correct device name. Previously it put an incorrect name of the block device in mnttab .
/etc/ttytype	This file is now included in the release tape.
/usr/lib/uucp/uusub	The -r option now reports send and receive counts correctly matching its headings. Previously the value was mistakenly swapped.
/etc/dump	Creates the file /etc/ddate if it does not exist and you are making a full dump.
setvbuf()	The parameters now match SVID. It is now setvbuf (stream, buf, type, size) . Previously it was setvbuf (stream, type, buf, size) .
longjmp()	Fixed longjmp to never return zero.

/bin/od

Previously this would crash if you tried to partition on a full optical disk.

/stand/fbackup

fbackup can now access multiple disk controllers. Previously, it would not back up any drive beyond number three.

6.5.1 ACP Problems Fixed In S/M/K1.7

- The problem with the ACP port hanging in several situations has been fixed. In some cases you may need to use the **stty flush** command or the new **stty reset** to clear a line that has hung due to excessive noise.
- The problem with interrupt sharing in the NOS ethernet driver has been fixed.

Chapter 7

ENHANCEMENTS

This chapter explains new features that have been added to Sys5 Release S/M/K1.7.

- Support is provided for larger buffer pools. However, setting NBUF to greater than 4000 may cause system crashes when the kernel runs out of address space.
- A new parameter in **conf.c** called SYNCRATE allows some control over the sync call made by **/etc/init** every 30 seconds. If SYNCRATE is left at zero as shipped, there will be no change from previous releases. However, setting SYNCRATE to a positive value will explicitly define the number of 1Kb buffers the system will flush to disk when **/etc/init** runs sync every 30 seconds. If this value is set low, delays to user processes when the specified number of buffers are being flushed will be very short or negligible, but the number of syncs necessary to flush the buffer pool will be increased. For example, a sync rate of 20 for a buffer pool of 400 blocks may cause user programs to delay about a second but could cause a modified 1Kb buffer to wait ten minutes before being written to disk. If a sync rate of 100 were chosen user programs could occasionally be delayed up to five seconds but a modified buffer would only remain in the system 2 minutes before being written to disk.

The suggested value for the SYNCRATE is 30 per disk drive, but this should only be set if the user notices the system appearing to pause significantly as it syncs the file systems. Setting SYNCRATE to 20 - 30 should alleviate this, but will keep data in the buffer pool, where it may be lost in a crash, or at very low values adversely affect the performance of the buffer pool.

Setting SYNCRATE reduces the number of unnecessary disk writes in temporary files, and also improves the effects of the buffer aging algorithms.

- The sync system call and sync command do not return until the disk I/O is complete. This does not affect the sync done every 30 seconds by **init**.
- Examples of how to effectively use a **ramdisk** are given in **/etc/rc**. Significant improvement can be made in the startup time for selected programs.
- This release supports the Televideo 970 terminal.
- The number of optical disk devices available has been increased to 64 from the previous limit of 32.

- The "L" option of the **dformat** standalone program is available on the P/95. This option recreates the defect list for the previously formatted drive.
- Minor device 255 for optical disk drivers can always be used to inquire about status.
- ACP and VCP throughput has been improved.
- A new option to **stty**, **-reset**, has been added.
- The new **rft** driver, **/dev/rft/device name**, no longer requires the definition of a disk partition to hold temporary data. The driver buffers information in memory and writes large blocks to cartridge to effect near streaming performance. Large blocks will be individually streamed to cartridge tape when IMSP disk activity is minimal. Output of large blocks in a streaming mode will improve cartridge tape performance in single user mode and idle multi-user mode.
- The configurable disk driver, **cdconf**, allows several logical disks to be joined into one logical disk. Can also be used to configure one minor device of the configurable disk driver or to reconfigure a device that was previously configured.
- The **cpio** utility now has a **size** option to specify the virtual size of each output media. The format is as follows:

cpio -o[acBv] [size[m|kib]]

The **size** option specifies the maximum number of bytes each output volume will hold. The **b** variable multiplies the size by 512, the **k** variable multiplies the size by 1024, and the **m** variable multiplies the size by 1024 * 1024.

CONFIGURABLE FILES

If you have previously made kernel modifications and relinked your kernel, or if the automatic patch relink fails, you will need to relink your operating system kernel manually.

In order to ease the generation of a new kernel, the internal table sizes and other portions of the kernel have all been parameterized. This, and the supply of system binaries (not source) allows us to distribute a system which is tailorable to a particular environment, while keeping within the terms of the licensing agreements.

8.1 Files In `/usr/src/uts/m68/cf`

In the directory `/usr/src/uts/m68/cf` you will find the following files:

- `conf.c` The main configuration file. This defines some system table sizes, and some parameters such as what device drivers are used in the system. It is here that the definition of device numbers is done.
- `linesw.c` The line discipline table. This is used by the terminal drivers. At present only one line discipline is allowed or used. It is possible that more will be added in the future.
- `low.s` This contains assembler source for the low level interrupt calls, the trap vectors, and other very low-level things for versions S and M.
- `klow.s` This contains assembler source for the low level interrupt calls, the trap vectors, and other very low-level things for version K.
- `name.c` Defines the local system name (for UUCP, TCP/IP and NOS), the current revision level of the kernel, and a kernel name (useful if you are booting more than one kernel). The kernel name is printed when a message like "name: panic: no fs" displays.

8.2 Inside conf.c

This section describes some of the constants that you might use to tune your system to your type of usage. The following is a (not necessarily complete) list of constants defined in the **conf.c** file.

- CDLIMIT** Defines how large the default maximum file size is. It should yield a long integer, such as `1L<<11`, which defines how many kilobytes the file may be. It is currently 2048Kb, allowing files to grow to 2Mb. This can be changed to suit your environment. Alternatively, if you do not have a release which implements this, you will need an **unlimit** program to allow you to create large files.
- MAXUP** Defines how many concurrent processes may be owned by a user-id (not super-user). If you have a common login for many users in an environment like data entry you may need to increase this to avoid a situation where a program may abort and display the message "cannot fork", because the current user-id has MAXUP processes already in the process table. It should not be larger than NPROC. The current setting allows any user-id to have 25 processes active.
- NOTE:** This limit does not affect the super-user.
- NBUF** Specifies how many system buffers to allocate. If you increase this, your disk read/write time will decrease, but the amount of memory available to the user will drop, so it is likely that more swapping needs to be done. However, if you only have a few users, a lot of memory, and disk speed is important, you can increase NBUF. This number should be a multiple of 4. Each entry in this table uses 26 bytes of memory, plus actual buffer space.
- NCALL** Determines the size of the system table that contains a list of functions that are to be invoked at some time (in 50ths of a second increments) in the future by the clock handler. The call-out table is used by terminal handlers to provide terminal delays and by various other I/O handlers. If this table overflows the panic message "timeout table overflow" will display, and your system will stop. Each entry in this table uses 6 bytes of memory.
- NCLIST** Specifies how many character list buffers to allocate. The buffers are linked together to form input/output queues for various character devices. The ICP kernel also contains some **clist** structures, which means this number is already about as small as it can get. Each entry in this table uses 28 bytes of memory.
- NFILE** Defines how large the file table is. Each entry in the table represents an opened file, either via **open**, **creat** or **pipe**. When this table is full, the diagnostic message **no file** displays, and no more files can be opened until some are closed. Each entry in this table uses 88 bytes of memory.
- NFLOCKS** Defines how many file locks can be active at any time. Remember that some files may have more than one region locked at a time, while others are never locked. This is a Plexus extension to standard *System V*, which did not support locking of files. Each entry in this table uses 10 bytes of memory.

- NMOUNT** Defines how many mounted file systems there may be at one time. The root file system (*/*) is always the first entry in this table. After this table is full, no more mounts can be done, and an error will be returned. Each entry in this table uses 8 bytes of memory.
- NPROC** Specifies how many process table entries to allocate. Each entry represents an active process. The swapper/scheduler is always the first entry, and **init** is always the second. This number depends on the number of terminal lines, and how many processes get created by each user. Each entry in this table uses 30 bytes of memory.
- NTEXT** Specifies how many text table entries to allocate. Each entry represents an active read-only text segment, which was created using the **-i** or **-n** option to the C compiler (**cc**) or link-loader (**ld**). When the table overflows, the diagnostic message "out of text" displays. Each entry in this table uses 12 bytes of memory.
- SMAPSIZE** Specifies how many entries there can be in the free-swap-blocks list. Each entry in this table represents a block on the swap device that is not in use. If the list overflows because this constant is too small, your system will crash in a very unpredictable manner. We recommend that you do not modify this constant from its current value. Each entry in this table uses 4 bytes of memory.
- SYNCRATE** Allows some control over the sync call made by **/etc/init** every 30 seconds. If SYNCRATE is left at zero as shipped, there will be no change from previous releases. However, setting SYNCRATE to a positive value sets a limit to the number of buffers written when **/etc/init** does its periodic sync. This does not affect the normal operation of the sync command or the sync system call. Setting SYNCRATE reduces the number of unnecessary disk writes of blocks in temporary files, and improves the effects of the buffer aging algorithms. The suggested value for the SYNCRATE is 30 per disk drive. There is more information about SYNCRATE in the *ENHANCEMENTS* chapter.

PLEXUS SOFTWARE SUPPORT POLICY

The Plexus Software Support Center offers software support for your Plexus software products. Please call the Support Center at 408-943-2250 for more information, or if you wish to order a support subscription.

9.1 Warranty Coverage

Plexus warrants that the tape will load and execute. Defective tapes will be replaced free of charge. Installation assistance can be purchased as an optional service.

9.2 Support Hours

Software Support is available weekdays from 6:30 A.M. to 5:00 P.M., Pacific Time.

Weekend service is not included in any of the support plans. Current rates for this premium service can be obtained from the Software Support Center. Service requested outside the standard schedule must be pre-arranged.

9.3 Preparation For Requesting Support

NOTE

About one quarter of the problems for which we receive calls are resolved in the manuals, so, be sure that you have read the documentation carefully.

Keep your documentation up to date and check the bulletin board.

It is helpful if you have the following information available before placing your call.

- Information about your hardware and software.
 - Machine serial number
 - CPU type
 - Software revision

- Memory size
- Applications
- Release level
- A detailed description of the failure symptoms, system environment, and all error messages. Be prepared to allow the analyst to dial in to your system.
- An authorization for service if your system is not under contract. (American Express number or Purchase Order number.)

PLEXUS COMPUTERS

Reader Comment Form

Company Name :

Your Name (Optional) :

Manual Name : **Plexus Sys5 UNIX S/M/K1.7 Release Notice**

Publication Number : **98-40199.7 Ver. A**

Please let us know if anything in this manual is unclear, incomplete, or inaccurate.

1. Should any information be included or removed?

2. Please specify the page and nature of any error(s) found in this document.

3. Other Comments

Please mail this form to :

Technical Publications Department
Plexus Computers, Inc.
3833 North First St.
San Jose, CA 95134



.

.



.

.

