Burroughs

XE 500 CENTIX™

Installation and Implementation Guide

Relative To Release Level 6.0
Priced Item
December 1986

Distribution Code SA
Printed in U S America
1207792

Burroughs

XE 500
CENTIX™

Installation and Implementation Guide

Relative To Release Level 6.0
Priced Item
December 1986

Distribution Code SA
Printed in U S America
1207792

NO WARRANTIES OF ANY NATURE ARE EXTENDED BY THIS DOCUMENT. Any product and related material disclosed herein are only furnished pursuant and subject to the terms and conditions of a duly executed Program Product License or Agreement to purchase or lease equipment. The only warranties made by UNISYS, if any, with respect to the products described in this document are set forth in such License or Agreement. UNISYS cannot accept any financial or other responsibility that may be the result of your use of the information or software material, including direct, indirect, special or consequential damages.

You should be very careful to ensure that the use of this information and/or software material complies with the laws, and regulations of the jurisdictions with respect to which it is used.

The information contained herein is subject to change without notice. Revisions may be issued to advise of such changes and/or additions.

Correspondence regarding this publication should be forwarded, using the Product Improvement Card at the back of this manual, or remarks may be addressed directly to UNISYS Corporation, Corporate Product Information East, 209 W. Lancaster Ave., Paoli, PA 19301, U S America

About This Guide

Purpose

This guide describes how to install XE 500 CENTIX software and implement the XE 500 CENTIX system.

Scope

This guide describes the procedures for installing XE 500 CENTIX system software. It also describes the tasks required to configure the software to match a system configuration.

Audience

The audience for this guide is

- The Burroughs field engineering representative, who will perform the initial software installation and implementation.
- The CENTIX administrator, who will perform the post-installation tasks and ongoing modifications to system configuration files as required.

Prerequisites

The system administrator who uses this guide should be familiar with the CENTIX operations discussed in the *XE 500 CENTIX Administration Guide* and the *XE 500 CENTIX Operations Guide*. The system administrator should also be familiar with the BTOS utilities, listed in the *XE 500 BTOS Operations Reference Manual*, and how to execute them.

How to Use This Document

Use this guide when installing XE 500 CENTIX system software and when configuring CENTIX system files.

Organization

This guide contains the following sections and appendices:

Section 1, Installation Overview, lists the major installation tasks and describes the types of release media. This section also discusses the role of BTOS in the CENTIX system.

Section 2, Starting Up the System, describes powering up and booting up the XE 500.

Section 3, Using Removable Media, explains proper procedures for handling disk cartridges and quarter-inch cartridge (QIC) tapes.

Section 4, Installing the Restricted Mode System Software, explains how to use the Boot Load utility to initialize the BTOS system disk and install a portion of the operating system that allows you to continue with installation.

Section 5, Installing XE 500 BTOS Software, explains how to use the MSysLoad utility and how to install standard XE 500 BTOS system software used in the CENTIX system.

Section 6, Creating the CENTIX Root Partition and Swap Files, describes how to use the Root/Swap utility to create these root partition and swap file for the initial software installation. It also describes how to move these partitions to other locations once software is installed.

Section 7, Installing the XE 500 CENTIX Software, explains how to use the CentixLoad utility and how to install standard XE 500 CENTIX system software.

Section 8, Supporting a Mixed Environment, outlines the installation steps for installing BTOS and CENTIX software to support both CENTIX and workstation BTOS.

Section 9, Installing CENTIX Environmental Software, provides an overview of the installation procedure for environmental software.

Section 10, BootLoadParams File, describes and gives the default values for the BootLoadParams file. This file defines how the BTOS system disk is to be initialized during Boot Load.

Section 11, Troubleshooting Software Installation Problems, lists error codes that can occur and describes the log files that are kept during installation. It also includes the procedure for correcting the problem caused by an uninitialized system disk.

Section 12, Overview of System Configuration Files, describes the text files that define the XE 500 BTOS hardware and software configuration.

Section 13, Overview of BTOS System Services, describes the services that control access to software applications and peripheral devices.

Section 14, Configuring BTOS System Services, explains how to use the MBtos Config utility to configure the XE 500 BTOS system services.

Section 15, Configuring CENTIX Software, outlines the tasks required to configure the CENTIX system.

Section 16, Post-Configuration Tasks, outlines the tasks that you should perform after the configuration procedures.

Section 17, Configuring the BTOS AdminAgent, explains the different modes in which the AdminAgent can be run. The AdminAgent controls the execution of utilities in XE 500 BTOS processors.

Section 18, Configuring the BTOS Queue Manager, explains how BTOS queues are created, and how to install and remove the Queue Manager. The Queue Manager controls the various queues in which related tasks are stored for processing (for example, printing queues).

Section 19, Configuring the lpr Printer Spooler, describes the BTOS-based lpr printer spooler and how to configure it into the system.

Section 20, Creating I/O Device Configuration Files, explains how to create and modify the BTOS files that define the hardware and software operating parameters of printers, tape drives, and communication devices such as modems and RS-232-C serial terminals.

Appendix A, Software Index, lists the CENTIX and BTOS files that are installed during the installation procedure.

Appendix B, Default System Configuration Files, show the default system configuration and initialization files for the standard XE 500 CENTIX.

Appendix C, Hardware Configuration Information, includes figures and tables that define disk naming conventions, processor naming conventions, and processor slot number conventions.

Appendix D, New CENTIX Device Naming Conventions, describes the new conventions for naming CENTIX devices as of the 6.0 release, and their correspondence to the pre-6.0 release conventions.

A glossary and index follow Appendix D.

Results

The Burroughs field engineering representative will be able to install and configure the XE 500 CENTIX system software.

The XE 500 CENTIX system administrator will be able to perform all tasks required to modify system configuration files, configure CENTIX, and perform the post-installation tasks.

Related Product Information

XE 500 CENTIX Operations Guide

This guide provides information on performing those XE 500 CENTIX tasks which are routinely performed by anyone using the XE 500 system.

XE 500 CENTIX Administration Guide

This guide provides information on administering XE 500 CENTIX system.

XE 500 CENTIX Operations Reference Manual

This manual provides a comprehensive list of the commands which are available through CENTIX. It also includes a brief explanation of how to execute each command.

XE 550 System Capabilities Overview

This guide describes the XE 550 CENTIX system software and hardware features.

XE 500 BTOS Customizer Operations Guide

This guide describes how to create customized versions of the operating systems that run on XE 500 BTOS processors.

BTOS Status Codes Reference Manual

This manual lists the BTOS system status codes, describes the problem to which each code refers and, if applicable, suggests how to recover.

Conventions Used in This Manual

In procedures, data that you are to enter at the keyboard are shown between quotation marks (") or as indented text. The quotation marks themselves should not be entered unless specifically stated in the text.

Variables are shown in italics. For example, in the run statement

    $run *runfile, parameters*

runfile and *parameters* are variables. In the actual run statement, *runfile* is replaced by the file name for the desired run file and *parameters* are replaced by the parameters associated with that run file.

CENTIX commands are shown in boldface.

In BTOS command forms and system configuration file descriptions, optional fields and parameters are enclosed in square brackets. When these field names are mentioned in the text, they are enclosed in quotation marks, for example, "[Print file]".

Contents

Illustrations

Tables

Installation Overview

This guide is used to install and implement the XE 500 CENTIX system.

The software installation comprises six major tasks to be performed in the following order:

1 Load the restricted mode XE 500 BTOS and CENTIX software onto the XE 500 BTOS system disk using the Boot Load utility.

   The Boot Load utility loads a portion of the XE 500 software from the Boot Load release media onto the XE 500 BTOS system disk. The system can then be booted up from this software in a restricted mode.

   The restricted mode includes a menu-driven facility through which you can run the remaining installation procedures.

2 Load the standard XE 500 BTOS software onto the XE 500 BTOS system disk using the MSysLoad utility.

3 Create the CENTIX root and swap partitions on the desired disks.

4 Load the standard XE 500 CENTIX software onto the appropriate XE 500 disks using the CentixLoad utility.

5 Create BTOS-based system configuration files using the MBTOS Config utility. This utility allows you to match the system configuration files to your specific system configuration.

6 Perform the necessary CENTIX configuration and post-configuration tasks.

When you have finished the previously listed tasks, you must reboot the system to put the installation and configuration changes into effect.

Note: *Installing standard system software and creating the default configuration files should take approximately one hour.*

To install and customize a large system, the procedure could require a considerable amount of time, effort, and knowledge of BTOS and CENTIX.

Abbreviated Outline of Installation Procedures

Along with the actual installation procedure, this manual incorporates much background information about the various installation utilities and system configuration files. The following outline lists the steps of the installation procedure. Included are references to where these steps are described in this manual.

Note: *If you are installing software to support a mixed system (that is, to support CENTIX terminals and BTOS workstations), you cannot use the procedure outlined above. Instead, refer to Section 8 for an abbreviated version of the appropriate installation procedure.*

STEP 1: Install the Restricted Mode Software from the Boot Load Release Medium

Refer to the subsection "Executing the Boot Load Utility" in Section 4.

STEP 2: Install the BTOS System Software from the BTOS Release Medium

Refer to the subsection "Installing the BTOS System Software" in Section 5.

STEP 3: Create the Root Partition

Refer to the subsection "Creating the Root Partition" in Section 6.

STEP 4: Create the Swap File for AP00

Refer to the subsection "Creating the Swap File" in Section 6.

STEP 5: Install the CENTIX System Software from the CENTIX Release Media

Refer to the subsection "Installing the CENTIX System Software" in Section 7.

STEP 6: Create the BTOS System Configuration Files

Refer to the subsection "Creating the BTOS System Configuration Files" in Section 14.

STEP 7: Reboot the System in Normal Mode

Refer to the subsection "Rebooting the System after Running MBTOS Config" in Section 14.

STEP 8: Install CENTIX Environmental Software, Configure CENTIX, and Perform Post-Configuration Tasks

Sections 15 through 20 outline all of the tasks related to installing CENTIX environmental software, configuring the CENTIX system, configuring BTOS services, and other post-configuration tasks.

Types of XE 500 Software Release Medium

The XE 500 CENTIX system software is available in these types of release media:

- Two or more disk cartridges. One disk cartridge is used for the Boot Load utility, the others by the MSysLoad and CentixLoad utilities. The disks are identified as to the software that they contain.
- One disk cartridge and two or more half-inch tapes. The disk cartridge is used for the Boot Load utility. The tapes are used for the MSysLoad and CentixLoad utilities. The tapes are identified as to the software that they contain.
- Three or more half-inch tapes. One tape is used for the Boot Load utility, the other(s) for the MSysLoad and CentixLoad utilities. The tapes are identified as to the software that they contain.
- Three or more quarter-inch cartridge (QIC) tapes. One QIC tape is used for the Boot Load utility, the other(s) for the MSysLoad and CentixLoad utilities. The QIC tapes are identified as to the software that they contain.

Table 1-1 lists the names by which the XE 500 CENTIX release media is identified.

Table 1-1 **XE 500 CENTIX Release Media Names**

Media Type	Media Name	Contents of Media
8-User CENTIX		
Disk Cartridge	BLD	Boot Load software
	BTOS-D	XE 500 BTOS for CENTIX system
	BN8-D1	XE 500 CENTIX, disk 1
	BN8-D2	XE 500 CENTIX, disk 2
	BN8-D3	XE 500 CENTIX, disk 3
Half-Inch Tape with Disk Cartridge	BLT	Boot Load software
	BTOS-T	XE 500 BTOS for CENTIX system
	BN8-T	XE 500 CENTIX
Half-Inch Tape with Disk Cartridge	BLD	Boot Load software
	BTOS-T	XE 500 BTOS for CENTIX system
	BN8-T	XE 500 CENTIX
QIC Tape	BLC	Boot Load software
	BTOS-C	XE 500 BTOS for CENTIX system
	BN8-C	XE 500 CENTIX
16-User and 32-User CENTIX		
Disk Cartridge	BLD	Boot Load software
	BTOS-D	XE 500 BTOS for CENTIX system
	BNX-D1	XE 500 CENTIX, disk 1
	BNX-D2	XE 500 CENTIX, disk 2
	BNX-D3	XE 500 CENTIX, disk 3
Half-Inch Tape with Disk Cartridge	BLT	Boot Load software
	BTOS-T	XE 500 BTOS for CENTIX system
	BNX-T	XE 500 CENTIX
Half-Inch Tape with Disk Cartridge	BLD	Boot Load software
	BTOS-T	XE 500 BTOS for CENTIX system
	BNX-T	XE 500 CENTIX

Table 1-1 **XE 500 CENTIX Release Media Names (Cont.)**

Media Type	Media Name	Contents of Media
QIC Tape	BLC	Boot Load software
	BTOS-C	XE 500 BTOS for CENTIX system
	BNX-C	XE 500 CENTIX
CENTIX Development System		
Disk Cartridge	BND-D1	Development System, disk 1
	BND-D2	Development System, disk 2
Half-Inch Tape	BND-T	Development System
QIC Tape	BND-C	Development System

Installation Assumptions

The software installation procedures described in this guide assume that you are familiar with how to use removable media (that is, disk cartridges, half-inch tape, and/or QIC tapes) and, if loading software from half-inch tape, how to use the tape drive.

See Section 3 for instructions on how to use removable media. See the operations manual for your tape drive for instructions on how to load a half-inch tape and use the tape drive.

Also, the PT 1500 to be used during the installation procedure must be connected to the first Cluster Processor (CP00) in the XE 500.

If you are loading software from half-inch tape, the tape drive must be connected to the first Disk Processor (DP00) or, if your system does not have a DP, the first Storage Processor (SP00).

These terminal and tape drive restrictions are necessary because only the first processor of each processor type is recognized during system software installation. (This is because the system runs in a restricted mode during system software installation.)

Troubleshooting Installation Problems

All installation utilities keep logs of the installation operations that they control. These logs are used to record error conditions that may occur, as well as normal operations.

The log files and status messages generated during system software installation are described in Section 11. This section also includes the procedure to correct the problem caused by initializing a previously uninitialized disk during the Boot Load utility.

Updating Software from Previous Release

If you are updating your CENTIX system software from a previous release, you should be aware that your old CENTIX root will be overwritten as part of the installation procedure. This is because the BTOS system disk must be initialized as part of the Boot Load utility. The initialization of the BTOS system disk destroys the old CENTIX configuration file, [sys]<sys>ConfigUFS.sys.

Therefore, before you update your software, save off any nonsystem files in the root file system that you want to keep.

Note: *You should not try to restore your old BTOS or CENTIX customized system files. All previous customization of system files must be repeated for the 6.0 release update.*

The only system files that you can save and reuse with the new software are the BTOS-based queue index file ([sys]<sys>queue.index), the lpr printer spooler configuration file ([sys]<sys>splcnfg.sys), and the lpr printer configuration files. You can also save any CENTIX nonsystem files (for example, uucp files) that you have customized.

BTOS in the CENTIX System

The XE 500 CENTIX system is actually based on two operating systems: **CENTIX** and a Burroughs workstation operating system called **BTOS.**

All CENTIX-based system software and applications run on the Applications Processor (AP).

BTOS runs on all other processor types. A BTOS-based processor, the master processor (FP00 or DP00, depending on your type of system), is responsible for controlling the front panel and downloading the other processors with their operating systems. BTOS-based processors also control input/output (I/O) to the peripheral devices, such as disk drives, tape drives, terminals, printers, data comm lines, and so on.

To properly install and maintain the system, it is necessary for you to manage BTOS files and services in addition to CENTIX. This manual contains all of the procedures needed to install and configure the BTOS portion of the XE 500 CENTIX system.

The following subsections provide background information about BTOS utilities and file name formats.

Running BTOS Utilities from CENTIX

There are numerous BTOS utilities that can be invoked through CENTIX. These utilities are used to maintain the BTOS portion of your software.

A BTOS utility is invoked by using the CENTIX **ofcli** command, including the utility's run file specification in the **ofcli** command entry. BTOS utilities can also be invoked through a system administration facility, called centrEASE, by filling in the utility's corresponding command form.

Accessing BTOS utilities is described in your *XE 500 CENTIX System Administration Guide* and in the *XE 500 CENTIX centrEASE Reference Manual*. Descriptions of BTOS utilities are described in the *XE 500 BTOS Operations Reference Manual*.

BTOS File Name Format

Before discussing the BTOS file name format, it is helpful to understand how the BTOS file system is structured.

Once a disk drive is properly formatted to accept data, it is said to contain a volume. The term *disk drive* refers to the hardware device; *volume* refers to the complete BTOS file system unit of information stored on the disk. Each formatted disk in the system has a volume associated with it.

BTOS frequently allows the device name of the disk drive and the volume name to be used interchangeably when referring to the information stored on the disk.

For more detailed information about disk device and volume names than is provided here, refer to the *XE 500 CENTIX Administration Guide.*

Disk Drive Device Names

Burroughs assigns disk drive device names that correspond to the drive's physical location in the system.

XE 500 built-in disk drives are denoted by the device names d0, d1, d2, and so on, depending on the enclosure disk drive slot in which they are located.

Storage module device (SMD) drives are denoted by the device names s0, s1, s2, and so on, depending on their location in the XE 500 base enclosure or MD3 enclosure.

Appendix C has figures that show how XE 500 disk device names are assigned.

Note: *An XE 500 disk can be a disk cartridge, a built-in disk, or an Storage Module Device (SMD) disk.*

In this guide, a "built-in disk" is assumed to be a 5 1/4-inch hard disk controlled by a File Processor (FP).

In some base enclosure styles, SMD disks can also be built into the XE 500; however, they will be referenced as "SMDs." All SMDs, whether they are in an XE 500 base enclosure or an MD3 enclosure, are controlled by Disk Processors (DPs).

Volume Names

A volume name is assigned to a disk when it is initialized using the MIVolume utility.

The volume name can later be changed using the MChange Volume Name utility.

The BTOS File System and File Names

Unlike CENTIX, the BTOS file system has a three-level structure:

- **Volume**. A volume is associated with each initialized disk in the system.
- **Directory**. Each volume can contain one or more directories. All directories are at the same level (that is, a directory cannot contain another directory).
- **File**. Each directory can contain one or more files.

A full BTOS file name designation has the following format:

[*volname*]<*dirname*>*filename*

where

volname	is the name of the volume on which the file is stored. Normally, the name of the disk device associated with the volume can be used instead of the actual name of the volume.
dirname	is the name of the directory in which the file is stored.
filename	is the name assigned to the file.

Processor Naming Conventions

XE 500 processor names take the form *Xpnn*, where *Xp* is the processor type mnemonic and *nn* is the number of that type in the system. Table 1-2 lists the processor types and their two character mnemonic. A figure in Appendix C shows the processor board numbering scheme.

Table 1-2 **Processor Board Designations**

Mnemonic	Processor Type
AP	Applications Processor
FP	File Processor
DP	Disk Processor
CP	Cluster Processor
SP	Storage Processor
TP	Terminal Processor

Master Processor

Each XE 500 CENTIX system has a master processor. This processor is loaded with the master BTOS operating system when the system is started. The master processor then controls the loading of the other processors with their operating systems. It also manages the system's interface with the base enclosure front panel controls.

Depending on the type of base enclosure ordered with the system, the master processor is the first processor in the base enclosure, either FP00 or DP00.

Applications Processor Names

Although all Applications Processors are, in general, referred to as APs, the CENTIX documentation uses the following conventions to distinguish the three types of AP boards:

- "APl" refers to APs based on the 68010 microprocessor.
- "APll" refers to APs based on the 68020 microprocessor without the floating point coprocessor.
- "APll with floating point" refers to APs based on the 68020 microprocessor with the floating point coprocessor.

Starting Up the System

This section describes how to perform the following tasks:

- Use the XE 500 front panel controls.
- Power up and power down the XE 500 enclosures.
- Boot up the XE 500.

XE 500 Controls

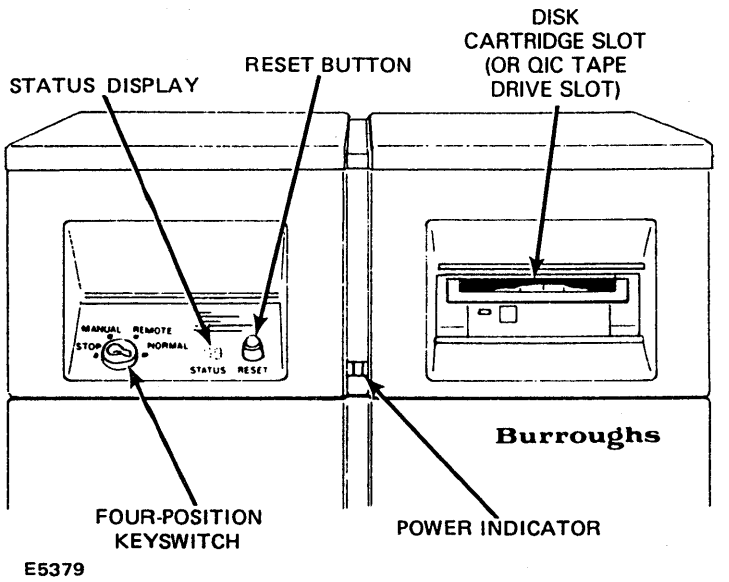
All of the XE 500 controls, except the POWER ON/OFF switch, are part of the base enclosure front panel.

The front panel, shown in Figure 2-1, is located just above the front door of the base enclosure.

The front panel consists of the following components:

- Four-position keyswitch.
- STATUS display.
- RESET button.
- Power indicator.
- Disk cartridge or quarter-inch cartridge (QIC) tape drive slot (optional).

Figure 2-1 **Sample XE 500 Base Enclosure Front Panel with Disk Cartridge Drive**



The keyswitch setting determines the operating mode of the XE 500. The control panel keys that were packed with the XE 500 are used to turn the keyswitch.

Select the operating mode by turning the key to the desired position. Lock the keyswitch at a selected position by removing the key.

The keyswitch positions are defined in Table 2-1.

Table 2-1 **Keyswitch Positions on the XE 500**

Keyswitch Position	XE 500 Operating Mode
STOP	Places the system in a reset state and prevents anyone from using the system.
MANUAL	Enables the RESET button and boots the system in manual mode.
REMOTE	Used during initial system software installation. Once software is installed, this keyswitch position is also used when troubleshooting system problems.
NORMAL	Disables the RESET button and boots the system in NORMAL mode.

Because having the keyswitch at MANUAL enables the RESET button, which could be accidentally pressed during system operation, it is recommended that you use the NORMAL keyswitch position. If you have to reset the system for any reason, you can do so by turning the key to STOP and then back to NORMAL.

The STATUS display shows status codes that indicate the operating status of the system.

The power indicator lights up when the system has been turned on and the internal power supply is operating properly.

Removable media can be used to load system software. They also provide a backup storage facility for the BTOS and CENTIX file system. Depending on the base enclosure model, your system may accept disk cartridges or QIC tapes. If the base enclosure does not have a disk cartridge or QIC tape drive slot, the removable medium for your system is half-inch tape.

Procedures for using QIC tapes and disk cartridges are given in Section 3. For half-inch tape handling instructions. Refer to the tape drive's documentation.

Powering Up the XE 500

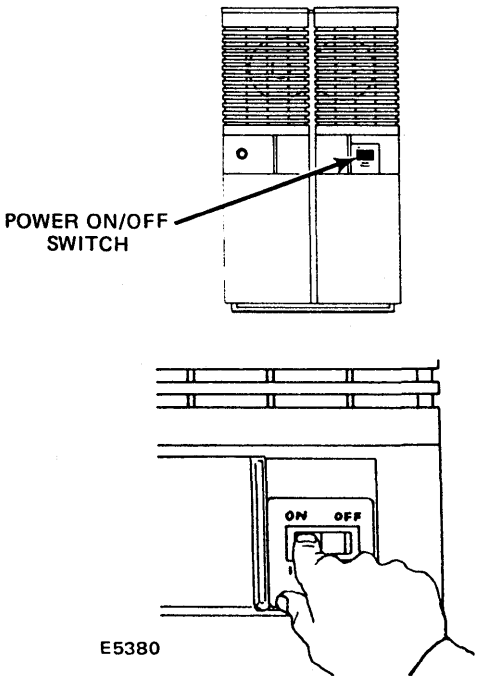
Power up the XE 500 system by performing the following procedure.

Caution: *Whenever you power up the XE 500, make sure that the removable medium drive is empty and the keyswitch is set to STOP.*

Referring to Figure 2-2, locate the POWER ON/OFF switch at the rear of each enclosure in the system.

Turn on power to the XE 500 by pressing the POWER ON/OFF switch of each enclosure to ON. When powering up enclosures in a multienclosure system, start with the base enclosure, then power up the second enclosure, the third enclosure, and so on.

Figure 2-2 **XE 500 Enclosure POWER ON/OFF Switch**



Once there is power to the system,

- The power indicator at the front panel of each enclosure should be lit.
- The STATUS display at the base enclosure should show the value "00."
- You should be able to hear the fans and feel a steady flow of air from the rear ventilation grille of each enclosure.

If any of these three conditions do not occur after turning on power to the XE 500, contact your Burroughs field engineering representative.

Once the enclosures are powered up and software has been installed, you boot (start) up the system by turning the keyswitch from STOP to MANUAL, REMOTE, OR NORMAL, depending on the operating mode in which you want the system to run. Refer to Table 2-1 and to the *XE 500 CENTIX Administration Guide* for more information on operating modes.

Powering Down the XE 500

Caution: *Whenever you power down the XE 500, be sure to first run the CENTIX* ***halt*** *or* ***shutdown*** *command to bring down the CENTIX system. Also be sure that the cartridge drive is empty and that the keyswitch is set to STOP. Always power down all other components in the system before powering down the XE 500 enclosure(s).*

To power down the XE 500, use the following procedure:

1 Make sure all users are logged off, and terminals and peripheral devices (that is, printers, tape drives, and so on) are turned off.

2 Make sure there is no removable medium currently mounted.

3 Turn the keyswitch to the base enclosure to STOP.

4 Power down each enclosure in the system, starting with the enclosure farthest from the base enclosure.

The power indicator(s) should go off, the STATUS display should go off, and the fans should stop operating.

Using Removable Media

Disk cartridges, QIC tapes, and half-inch tapes provide portable media on which to store information. In the XE 500 system, they can provide the source for the initial system software and for future software updates. They can also be used as backup storage devices for important files in the XE 500 system.

This section discusses how to use disk cartridges and QIC tapes. For half-inch tape handling information, refer to the tape drive's documentation.

Using Disk Cartridges

Handling Disk Cartridges

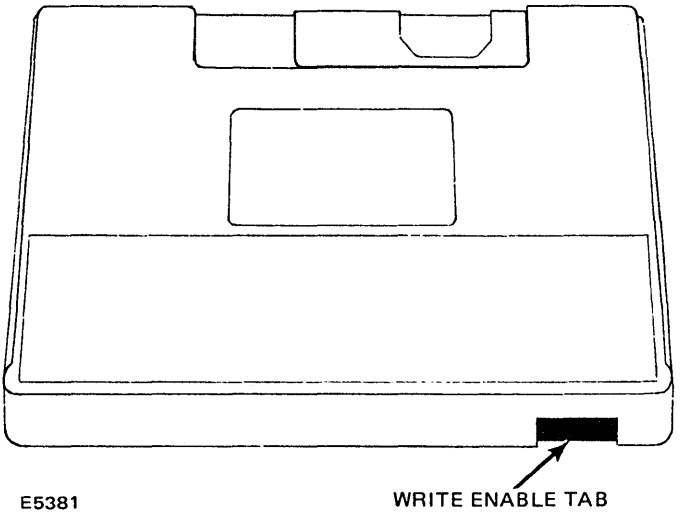
A disk cartridge consists of a rigid disk enclosed within a protective plastic cartridge. The disk is covered with magnetic material on both sides. Every disk cartridge comes with its own case, which serves as a dust jacket.

There are some simple rules to follow when you handle disk cartridges.

- Use care when handling disk cartridges. Sudden shocks can cause internal damage to the cartridge.
- Store disk cartridges upright in their dust jackets. Keep them out of direct sunlight. Store them in a dry area at normal room temperature.
- Store disks away from magnetic or electrical devices.
- Do not use disks that have recently been in an extremely hot or cold environment. Wait for them to adjust to room temperature.

Caution: *DO NOT remove the red write enable tab at the side of the disk cartridge. (See Figure 3-1.)*

Figure 3-1 **Disk Cartridge**



Operating the Cartridge Slot

You will need to know how to operate the cartridge slot when using disk cartridges. There are only two controls for the cartridge slot—the release latch and the slot door. Referring to Figure 3-2, locate these two parts of the cartridge slot.

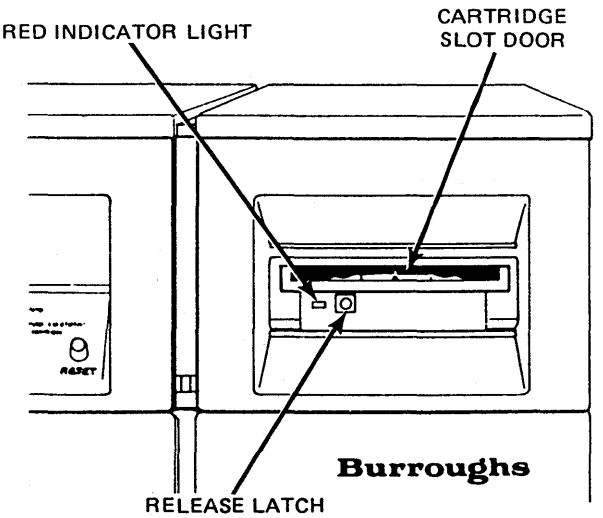
The release latch releases the door from the locked position. The door must be manually opened before you can insert or remove a disk cartridge.

Notice that there is a red indicator light to the left of the release latch. This light goes on whenever the system is retrieving or storing information on the disk cartridge.

While a disk cartridge is inserted, the disk spins at a certain operating speed. The indicator light flashes whenever the disk is being brought down from its operating speed.

Caution: *Never push the release latch when the indicator light is on. Also, you should never open the slot door to remove a disk cartridge while the light is flashing. These actions could damage the disk and/or the drive unit.*

Figure 3-2 **Disk Cartridge Slot**



Inserting Disk Cartridges

Caution: *The XE 500 base enclosure should always be powered up before a disk cartridge is inserted.*

Use the following procedure to insert a disk cartridge.

1 Be sure that the XE 500 base enclosure is powered up.

2 Press the release latch at the cartridge slot.

3 Open the cartridge slot door.

Caution: *To avoid internal damage to the cartridge drive when inserting a disk cartridge, be sure that the cartridge slot door is fully open.*

4 Remove the disk cartridge from its dust jacket.

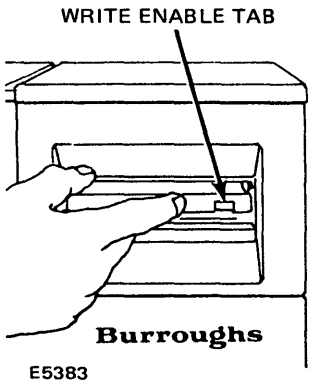
5 Insert the disk cartridge as shown in Figure 3-3. The side that has the write enable tab should be facing out, and the write enable tab should be to the right.

6 Push the disk cartridge into the drive until you hear a "click."

7 Close the door. Note that the red indicator lights up when the disk cartridge drive is being used.

Caution: *Do not turn off power to the XE 500 base enclosure after a disk cartridge has been inserted.*

Figure 3-3 **Inserting the Disk Cartridge**



Removing a Disk Cartridge

Caution: *Attempt to remove a disk cartridge only when the disk drive indicator light is off. Never press the release latch when the light is on or blinking.*

Use the following procedure to remove a disk cartridge.

1 Make sure the disk drive indicator light is off.

2 Press the release latch.

After the cartridge release latch is pressed, the indicator light at the cartridge slot blinks for about 30 seconds (until the disk stops spinning).

Caution: *After pressing the release latch, do not open the drive door until the blinking stops.*

3 Be sure that the indicator light has stopped blinking.

4 Open the cartridge slot door.

As you fully open the door, the disk automatically ejects from the drive. Hold onto the disk during this process to prevent it from falling.

5 Place the disk cartridge into its dust jacket.

6 Close the cartridge slot door.

Note: *Always keep the cartridge slot door closed when not in use.*

Using QIC Tapes

Handling QIC Tapes

Here are some rules to follow when handling QIC tapes:

- Use care when handling QIC tapes. Do not touch or manually move the magnetic tape inside the cartridge.
- Store cartridges in their cases, in a dry area at room temperature. Unlike half-inch tapes, it is not necessary to store QIC tapes vertically.
- Keep QIC tapes away from magnetic devices, such as CRT screens.

Under certain conditions, the QIC tape should be *packed* (that is, rewound) to ensure that the tape operates properly during I/O operations. Before using a QIC tape, execute the BTOS MQic Retension utility to rewind the tape if any of the following conditions apply:

- Occurrence of excessive read/write errors (more than 50 software errors per pass).
- Exposure of the cartridge to temperatures outside the range of 40°F to 110°F (5°C to 44°C).
- Prolonged storage (more than two weeks) of the cartridge.
- Physical shock to the cartridge, such as dropping it or dropping something on it.

For information on executing the BTOS MQic Retension utility, refer to the *XE 500 BTOS Operations Reference Manual*. For information about executing BTOS utilities from CENTIX, refer to the *XE 500 CENTIX Administration Guide*.

Figure 3-4 shows the components of a QIC tape. Figure 3-5 shows the positions of the write protect plug for enabling and disabling write operations to the tape.

Figure 3-4 **Components of a QIC Tape**

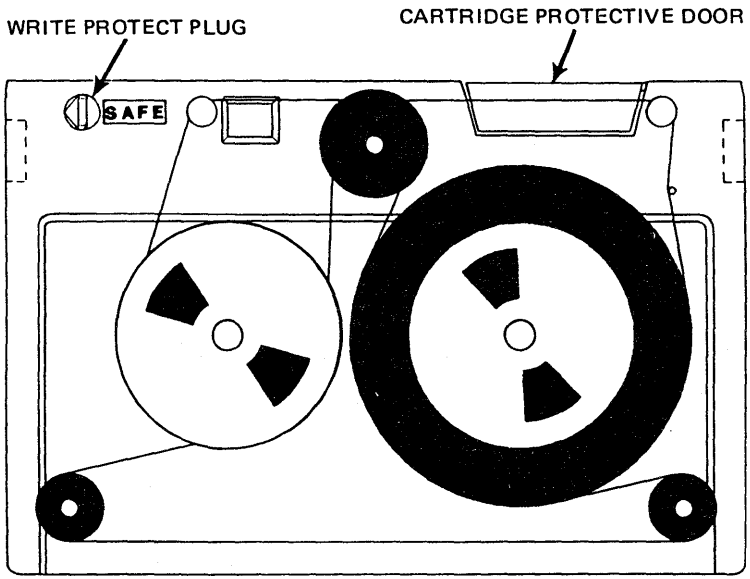
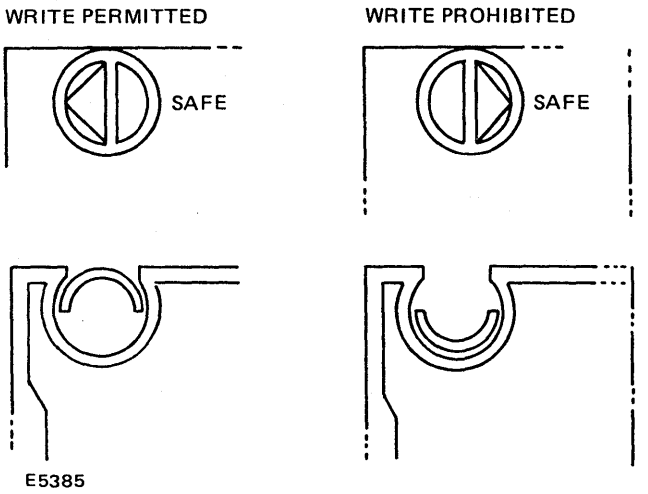


Figure 3-5 **QIC Tape Write Protect Plug Positions**



Operating the QIC Tape Drive

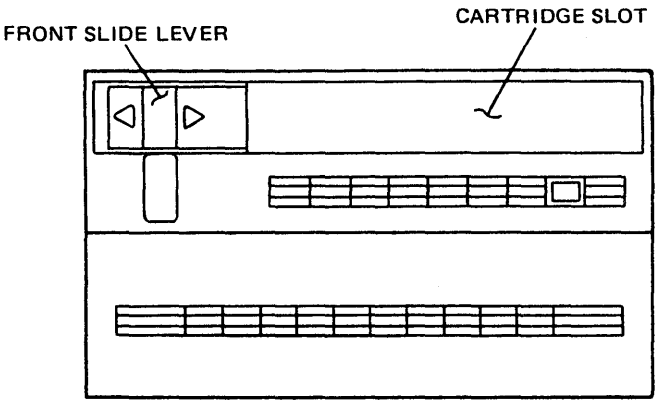
You will need to know how to operate the QIC tape drive when using QIC tapes. There are only two controls for the drive—the front slide lever and the cartridge slot. Referring to Figure 3-6, locate these two parts of the drive.

The front slide lever controls the locking of the QIC tape in the drive.

Notice that there is a red indicator light on the right side of the drive. This light goes on whenever the system is retrieving or storing information on the drive.

Caution: *If you push the front slide lever when the indicator light is on, you could interrupt a write operation to the tape.*

Figure 3-6 **The QIC Tape Drive**



Inserting a QIC Tape

Use the following procedure to insert a QIC tape.

1 Be sure that the XE 500 base enclosure is powered up.

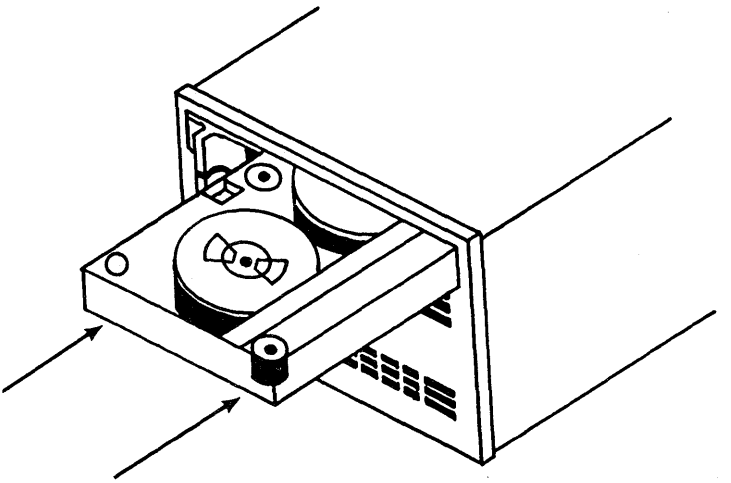
2 Remove the QIC tape from its case.

3 Insert the QIC tape into the drive with the cartridge protective door facing left and the base plate of the cartridge facing down, as shown in Figure 3-7.

4 Push the QIC tape into the drive.

5 When the QIC tape is fully inserted, it descends into the tape drive. At this time, the protective door on the cartridge opens.

Figure 3-7 **Inserting the QIC Tape**



E7517

6 Move the front slide lever to the right until it reaches the lever stop. This secures the cartridge and brings the head assembly to its correct operating position.

7 The QIC tape is now loaded. Note that the red indicator lights up when the QIC tape drive is being used.

Removing a QIC Tape

Use the following procedure to remove a QIC tape.

1 Make sure the drive indicator light is off.

2 Move the front slide lever to the left until it reaches the lever stop.

3 The head assembly in the drive retracts, and the protective door on the cartridge closes. A cartridge ejector automatically raises the cartridge out of the drive and slowly pushes it forward.

4 Grasp and remove the QIC tape, and return it to its case, with the protective door to the inside of the case and the base plate down.

Installing the Restricted Mode System Software

The restricted mode software is the portion of the XE 500 software that you install first into the system. Once installed, the restricted mode software allows you to boot up the system, install the rest of the software, and configure the system. The restricted mode software can also be used to recover from a system software failure due to the corruption of system files. Refer to the *XE 500 CENTIX Administration Guide* for a complete description of what you can do using the restricted mode.

The Boot Load utility is used to initialize the XE 500 BTOS system disk (disk d1 or s0), if needed, and to install the restricted mode software. The utility accepts the XE 500 front panel keyswitch positions as input.

Initializing the BTOS System Disk

The first part of the Boot Load procedure includes a choice of initializing or not initializing the BTOS system disk (disk d1 or s0) before installing the restricted mode software.

You **should initialize** the BTOS system disk

- If this is the first time 6.0 release system software is being installed.
- If corrupted system software has prevented the system from running and you are reinstalling the system software or restoring system software from archive media.

You can choose not to initialize the system disk if you have already installed the XE 500 CENTIX 6.0 release and you are only adding new BTOS software products or BTOS software products that you left out of previous installation procedures.

Executing the Boot Load Utility

Note: *Once restricted mode software is installed, you can edit the BootLoadParams file to modify how the BTOS system disk is initialized. See Section 10 for details.*

Use the following procedure to execute the Boot Load utility. This procedure installs the restricted mode software onto the system disk.

1 Power up the XE 500 and set the base enclosure keyswitch to STOP.

2 Turn on one of the PT 1500s connected to the first CP board (CP00). All other PT 1500s that are connected to CP00 must be off.

3 Insert the Boot Load release medium into the drive (refer to Section 3 for details on using removable media).

If you want to initialize the system disk before installing system software, go to step 4.

If you do not want to initialize the system disk before installing system software, go to step 6.

4 Turn the keyswitch to REMOTE to initialize the BTOS system disk.

The system boots from the release medium, displaying the normal sequence of status codes (01-20) in the front panel STATUS display.

Once the Boot Load utility begins to execute, the status code 80 is displayed.

5 When the status code 81 is displayed, the utility pauses, waiting for new input from the keyswitch. You can choose to initialize or not initialize the system disk at this point.

To initialize the system disk, turn the keyswitch to NORMAL. The status code 82 is displayed, indicating that the BTOS system disk is being initialized. When disk initialization is complete, the status code 86 is displayed. Go now to step 7.

If you decide not to initialize the system disk, you can terminate the current session of Boot Load as long as an 81 is displayed on the front panel. See the next subsection, "Stopping the Initialization of the System Disk."

Caution: *Do not turn the Keyswitch to STOP while an 82 is displayed.*

---

6 Turn the keyswitch to NORMAL to start the Boot Load utility.

7 The utility begins loading software to a fixed disk, displaying status codes in the range of 86 to 89.

The utility indicates the successful completion of the software loading operation by displaying a 90.

A status code in the range of 92 to 99 indicates that an error has occurred and an error message has been logged in the BTOS file [sys]<sys>BootLoad.log. In such a case, refer to Section 11.

8 The Boot Load utility is now complete. Turn the keyswitch to STOP and remove the release medium.

9 Reboot the system by turning the keyswitch from STOP to REMOTE.

A banner is displayed at the terminal that tells you the system is running in the restricted mode. The system then prompts you to fill in the correct date and time.

10 Fill in the date and time and press the RETURN key.

The date and time should be entered in the format shown on the screen.

11 After the correct date and time have been entered, the system checks the two CENTIX file systems used by the restricted mode software, /dev/root and /dev/fpSAFrestore.

If the check fails, the system will prompt you to run the CENTIX **fsck** (file system check) command manually. The system will then run the superuser shell, denoted by the pound sign (#) prompt.

Enter the two fsck commands as follows:

```
# fsck /dev/root
# fsck /dev/fpSAFrestore
```

When these file systems have been successfully checked, enter

```
# exit
```

to exit from this shell. The system then displays the Restricted Mode Main Menu.

Caution: *Do not try to execute any other CENTIX commands from this shell. It is intended strictly for running* ***fsck*** *on the two restricted mode file systems.*

If the manual file check is not successful, reinstall the restricted mode software again.

---

12 You are now ready to install your BTOS software. Go to Section 5.

Note: *As long as a Boot Load medium is not on-line, you will now always be able to bring the system up in restricted mode by turning the keyswitch from STOP to REMOTE. Refer to your* XE 500 CENTIX Administration Guide *for a complete description of what you can do using the restricted mode.*

Stopping the System Disk Initialization

If after turning the keyswitch to REMOTE, you decide that you do not want to initialize the system disk, use the following procedure to terminate the Boot Load utility.

Caution: *After starting an installation procedure with the keyswitch at REMOTE, you can only terminate the installation while an 81 code is being displayed.*

DO NOT TURN THE KEYSWITCH TO THE STOP POSITION WHILE AN 82 APPEARS IN THE STATUS DISPLAY.

If disk initialization is interrupted (that is, turning the keyswitch to STOP while an 82 is displayed), the badblk.sys file, which contains bad spot information, will be corrupted and Boot Load will be halted.

If you do turn the keyswitch to STOP during disk initialization, see "If You Interrupt Disk Initialization" in Section 11 for instructions on how to recover.

---

1 With the STATUS display showing 81, turn the keyswitch to MANUAL.

  This action will abort the current session of SysLoad and display a termination status of 92.

2 Once the termination status 92 is displayed, turn the keyswitch to STOP.

3 To begin software installation again, refer to the appropriate installation procedure.

Logging the Status of the Boot Load Process

When the Boot Load utility is finished, the Boot Load session just completed is recorded in a BTOS log file, [sys]<sys>BootLoad.log.

When running Boot Load from a disk cartridge, the first time that the utility is run the log file is stored in [d0]<sys>BootLoad.log and copied to [sys]<sys>BootLoad.log. Updates are appended to [sys]<sys>BootLoad.log.

When running Boot Load from tape, the first time that the utility is run the log file [sys]<sys>BootLoad.log is created. Because it cannot be created until the system disk is initialized, it does not contain the BTOS MIVolume utility status report. Updates are appended to [sys]<sys>BootLoad.log.

The log provides a record of the software that was installed during the session. Also, any installation error messages that occurred are contained in this file. The session is denoted in the file by a banner that indicates the Boot Load utility.

See Section 11 for a detailed description of the log file.

Installing XE 500 BTOS Software

Once the restricted mode software installation is complete, you can install the XE 500 BTOS system software onto the BTOS system disk. This is done at the CENTIX terminal through the interactive MSysLoad utility.

The MSysLoad utility provides a menu-driven facility through which you can

- Install all of the standard BTOS system software from the release media.
- Install only selected products of the BTOS standard system software from the release media.

The MSysLoad utility can also be used to reinstall 6.0 release BTOS system software.

If an error occurs during the installation process, an error menu is displayed. This menu allows you to exit the MSysLoad utility or to continue installing BTOS products. The product causing the error is not installed.

The MSysLoad utility also logs the user interactions and system operations that occur during the session in a BTOS log file, [sys]<sys>SysLoad.log.

Refer to Section 11 for a complete description of the MSysLoad error menu and log file.

Note: *After completing an MSysLoad utility session, you should always print the log file and then remove it to save system disk space.*

This section describes the entire installation procedure, including a detailed description of the MSysLoad utility.

Checking for Improperly Formatted System Disk

Prior to initializing the BTOS system disk, the Boot Load utility checks the system disk device to see if it has already been initialized. If it has, the Boot Load utility can use information in the disk's volume control structure files to determine what type of disk device it is. The utility can then reinitialize the disk with the proper parameters.

If the disk has not been initialized before being installed into the XE 500, the Boot Load utility cannot tell what type of device it is. The Boot Load utility initializes the disk with a default set of device parameters. These parameters cause the disk to be initialized for only about 17 MB, which is lower than the amount of memory space actually available on the disk.

The procedure described in the next subsection includes steps to check that the disk has been initialized for its fullest capacity.

If you find the disk has not been fully formatted, refer to the recovery procedure provided in Section 11.

Installing the BTOS System Software

The installation procedures for disk cartridges and tapes are identical. The procedure assumes that you have completed the restricted mode installation instructions described in Section 4. The restricted mode's Main Menu should be displayed at the terminal.

1 The Restricted Mode Main Menu and prompt are displayed as follows:

```
        RESTRICTED MODE MAIN MENU

.....................................
**      0. Shutdown                  **
**      1. Restore                   **
**      2. Set up Software           **
**      3. Go into Shell             **
.....................................

Please Select Option ==>
```

2 Enter "2" and press the RETURN key. The Set Up Software Menu is displayed.

3 The Set Up Software Menu and prompt are displayed as follows:

```
          SET UP SOFTWARE MENU

**************************************
**     0. Return to Main Menu       **
**     1. Run MSysLoad              **
**     2. Set Root and Swap Location **
**     3. Run CentIxLoad            **
**     4. Run MBTOS Config          **
**************************************

Please Select Option ==>
```

Enter "1" and press the RETURN key. This causes the MSysLoad utility to begin. The MSysLoad Main Menu is displayed.

Note: *If you need detailed information about using the MSysLoad utility, refer to the next subsection, "Using the MSysLoad Utility."*

4 At the Main Menu prompt, enter the number corresponding to the installation method you want to use (that is, load all BTOS software automatically or selectively load software products). Then press the RETURN key.

The Media Menu is displayed.

5 Load the appropriate BTOS software release medium (BTOS-D for a disk cartridge, BTOS-T for a half-inch tape, BTOS-C for a QIC tape).

6 At the Media Menu prompt, enter the number corresponding to the medium type and press the RETURN key.

If you selected to load all software automatically, the utility displays a message indicating when the installation operations are complete.

If you are selectively loading software, the utility displays the Load Software Menu for each product not required to run the system. This menu allows you to choose whether you want to install the optional software product. If a product is required to run the system, it is installed automatically. When the utility has cycled through all products on the release medium, it displays a message indicating that the installation operations are complete.

7 When the release medium has been processed, you are prompted to press the GO key to return to the MSysLoad Main Menu. Press the GO key.

8 At the MSysLoad Main Menu prompt, enter "0" and press the RETURN key.

You are returned to the restricted mode's Set Up Software Menu.

9 At the Set Up Software Menu prompt, enter "0" and press the RETURN key.

You are returned to the Restricted Mode Main Menu.

10 Enter "3" and press the RETURN key. The CENTIX superuser shell prompt (#) is displayed.

11 Following the shell prompt, enter

```
# ofcli
```

and press the RETURN key. The CLI prompt ($) is displayed.

12 Following the CLI prompt, enter

```
$run [sys]<admin>MVolumeStatus.run,dd
```

where *dd* is "d1" for a system with a master FP and "s0" for a system with a master DP. Then press the RETURN key.

The volume status report for the XE 500 BTOS system disk is displayed at the terminal screen.

13 Note the value given for free sectors.

If the value is approximately 10000 sectors, or if it is not close to the expected value for the device type of the system disk minus approximately 20000 sectors, do not continue with this procedure. Instead, go directly to the procedure outlined in Section 11 for recovering from an improperly formatted BTOS system disk.

If the value is close to the expected value for the device type of the system disk, go to step 14.

14 Following the volume status report, the CENTIX superuser shell prompt (#) is displayed. Enter

```
# exit
```

and press the RETURN key. The Restricted Mode Main Menu and prompt are displayed again.

15 Enter "2" and press the RETURN key. The Set Up Software Menu is displayed.

16 You are now ready to create the root and swap partitions.

Go now to Section 6 for instructions on how to create the root and swap partitions.

Using the MSysLoad Utility

The MSysLoad utility is a menu-driven facility through which you install BTOS software products from release media.

Each MSysLoad menu comprises

- A heading that describes the purpose of the menu.
- A list of actions from which to select. Each action is assigned a number, which is used to enter your selection.
- A prompt at which you enter the number of the action you want to execute. After entering the desired number, you must press the RETURN key.

The Main Menu provides the option of leaving the utility and returning to the restricted mode Set Up Software Menu. All other menus provide the option of returning to the Main Menu. Therefore, you are always no more than one menu away from being able to terminate the utility. You can also terminate the utility at any time by pressing the FINISH key.

Main Menu

When the utility is invoked, a welcome message is printed along with the Main Menu. The Main Menu gives you the option to

- Exit the utility and return to the BTOS Executive.
- Automatically install all of the software.
- Selectively install files from the software.

The Main Menu and prompt are displayed as follows:

```
          WELCOME TO THE SYSTEM LOAD UTILITY

..................................................
..                                              ..
..        Sysload Install Utility Main Menu     ..
..                                              ..
..                                              ..
..     0 - Exit the System Load Utility         ..
..                                              ..
..                                              ..
..     1 - Install all Software Automatically   ..
..                                              ..
..     2 - Selectively Install Software         ..
..                                              ..
..................................................

      Please Select Method of Installation ==>
```

Note: *If you are reinstalling 6.0 release system software, you will probably want to select option 2.*

To exit from the utility, enter "0" and press the RETURN key. The message "System Load Utility Complete" is displayed and the command line is returned to the screen.

To automatically install all software on the medium, enter "1" and press the RETURN key. To selectively install software on the medium, enter "2" and press the RETURN key.

After selecting either option 1 or 2, the Media Menu is displayed. The Media Menu allows you to identify the type of release medium from which your software is installed.

Media Menu

The Media Menu and prompt are displayed as follows:

```
                    Media Menu

.....................................................
..                                                   ..
..  What medium are you using to install software?   ..
..                                                   ..
..                                                   ..
..  0 = Return to Main Menu                          ..
..                                                   ..
..                                                   ..
..  1 = Medium is Disk Cartridge                     ..
..                                                   ..
..  2 = Medium is Half Inch Tape                     ..
..                                                   ..
..  3 = Medium is Quarter Inch Cartridge (QIC Tape)  ..
..                                                   ..
.....................................................

Please Select Medium Type ==>
```

Enter the number corresponding to the medium type and press the RETURN key.

If you have not already inserted your medium into the drive, a message will be displayed telling you that the utility is waiting for the medium to become ready. Insert your medium.

Note: *The utility waits 30 seconds for the medium to come on-line, then displays a message that the medium is not mounted. The utility then waits about a minute; if the medium is still not on-line, you are prompted to return to the Main Menu.*

If you originally chose to load all system software (option 1 of the Main Menu), then, after you specify the medium type and insert your disk or tape, MSysLoad begins installing products. As products are installed, the utility displays messages that indicate the various installation operations being performed (see the subsection "Product Installation Messages" for an explanation of the messages).

If you originally chose to load system software selectively (option 2 of the Main Menu), then, after you specify the medium type, MSysLoad searches for the first product on your medium and displays the Load Software Menu.

Load Software Menu

The Load Software Menu and prompt are displayed as follows:

```
                Load Software Menu

..................................................
..                                              ..
..      Do you want to load this software?      ..
..                                              ..
..                                              ..
..      0 = Return to Main Menu                 ..
..                                              ..
..                                              ..
..      1 = Load This Software                  ..
..                                              ..
..      2 = Skip This Software                  ..
..                                              ..
..................................................

The Product is: productname

  Please Select Whether to Load or Skip ==>
```

To return to the Main Menu, enter "0" and press the RETURN key. If the medium is tape, MSysLoad will rewind and unload the tape.

To install the product, enter "1" and press the RETURN key. MSysLoad installs that product. As the product is installed, the utility displays messages that indicate the various installation operations being performed (see the subsection "Product Installation Messages" for an explanation of the messages). When the product has been installed, MSysLoad displays the Load Software Menu again for the next product on the release medium.

To skip installation of the product, enter "2" and press the RETURN key. MSysLoad displays the Load Software Menu again for the next product on the release medium.

The Load Software Menu is displayed for each product on the release medium that is not required to bring up the system. If MSysLoad finds a product that is required to bring up the system, it loads that product automatically while displaying the message

```
** This software is needed to bring up the system.
   It will be loaded automatically.
```

For some products, the MSysLoad utility will display additional comments below the product name. These comments are provided to help you determine whether you want to install the product or skip to the next product.

When the Medium Has Been Processed

When MSysLoad has cycled through all of the products on the release medium, a message indicating the completion of the medium's software installation is displayed. You are prompted to press the GO key to return to the Main Menu, from which you can choose to install more software products or exit the utility.

If you choose to exit the MSysLoad utility, the following exit message is displayed:

```
System Load Utility Complete.
```

Product Installation Messages

While a product is being installed, messages are displayed at the screen to describe the operations being performed by the utility. These operations can include

- Copying files from the release medium to the system disk.
- Appending data to files already on the system disk.
- Creating directories on the system disk.
- Creating files on the system disk.
- Appending submit text.
- Error messages.

Creating the CENTIX Root Partition and Swap File

This section describes the Root/Swap utility. If you are installing the CENTIX software for the first time, perform the procedures in the subsections "Creating the Root Partition" and "Creating the Swap File." Then go to Section 7 for instructions on installing CENTIX system software.

The Root/Swap utility is an interactive utility through which you create the root partition and the swap file for AP00 as part of an initial software installation. The Root/Swap utility can also be used to replace an existing root partition, or to move or change the size of existing root partition and swap file.

The Root/Swap utility keeps a log, /rootswap.log, of all user interaction and system operations that occur during a session. This log file is kept in the restricted mode CENTIX root file system. Refer to Section 11 for a complete description of the Root/Swap utility log.

Caution: *The Root/Swap utility will not stop if a disk I/O error occurs during the session. Therefore, you should run the BTOS utility MPLog after completing a Root/Swap utility session to make sure that no disk I/O errors occurred during the session.*

You should also check the Root/Swap utility log to make sure that all other utility operations ran successfully before continuing with the installation procedure.

Note: *After completing a Root/Swap utility session, you should always print the log file. This log file is automatically removed the next time the Root/Swap utility is initiated.*

Once you have installed your 6.0 release CENTIX system software, you can replace your root (refer to the subsection "Replacing a Root Partition"). You can also move or change the size of your root partition or swap file (refer to the subsections "Changing the Root Partition," and "Changing a Swap File").

Note: *The Root/Swap utility can only create the swap file for the first AP in the system (AP00).*

Swap files for other APs in the system are created when you use centrEASE to configure the other APs into the system.

Configuring other APs into the system is discussed in Section 15.

Overview of Root Partition and Swap File

In the CENTIX hierarchal file system structure, the top directory is the **root** directory. Most of the CENTIX system software is stored in the file system associated with the root directory.

CENTIX requires a **swap** area on one of the XE 500 disk storage devices. The swap area is used to temporarily store the jobs that are not currently being processed. The AP *swaps* jobs between this swap area and local onboard memory to carry out the multitasking function. There must be one swap area for each AP in the system.

Every CENTIX file system, including the root, require a *BTOS partition* on a disk device in which to be stored. From the perspective of BTOS, a partition is simply a large file stored in the <sys> directory of a disk. The BTOS file name for a partition is "partition.*n*," where *n* is the partition number (0-7).

Swap files are also in the BTOS <sys> directory of the volume on which they are stored. The BTOS file name for a swap file is Swap.Ap.*nn*, where *nn* is the number of the corresponding AP board.

In CENTIX, each partition file is assigned a minor device number and is associated with a particular file system through the **mount** command. Each swap file is also assigned a minor device number.

How the Root/Swap Utility Works

The Root/Swap utility is invoked from the Setup Software menu of the restricted mode CENTIX.

Upon being invoked, the Root/Swap utility searches for the BTOS-based CENTIX configuration file, [sys]<sys>ConfigUFS.sys. This configuration file contains entries that assign CENTIX minor device numbers to all of the possible BTOS partitions and the swap files.

If the Root/Swap utility does not find this file, it assumes that CENTIX software has not been installed yet. The restricted mode's CENTIX configuration file, [sys]<sys>ConfigUFS.r.sys, is copied to create [sys]<sys>ConfigUFS.sys. The Root/Swap utility then runs a series of menus and prompts that allow you to create the root partition and swap file.

If the Root/Swap utility does find this file, it assumes that CENTIX software has already been installed and that the root partition and swap file already exist. It then runs a series of menus and prompts that allow you to create a new root partition to replace the old one, or to move or change the size of the existing root partition and swap files. The Root/Swap utility knows where the current root partition and swap files are located by reading the appropriate entries in the CENTIX configuration file.

Terminating a Root/Swap Utility Session

You can terminate the Root/Swap utility at any time by pressing the DELETE key, except when the following system operations are being executed:

- Creating the root partition.
- Making the file systems for the root partition.
- Executing the **mklost+found** command on a new root partition.
- Creating the swap file.
- Initializing a disk.
- Updating the CENTIX configuration file.

Creating the Root Partition

After completing the BTOS software installation procedure described in Section 5, the restricted mode's Set Up Software Menu should be displayed at the PT 1500. Perform the following procedure to create the root partition.

1 In response to the Set Up Software Menu prompt, select the Root/Swap utility by entering "2" and then press the RETURN key.

   The system prompts you for whether or not you want the root partition on the default disk.

2 To put the root partition on the default disk, enter "y" and press the RETURN key. Then go to step 4.

   To put the root partition on another disk, enter "n" and press the RETURN key. Then go to step 3.

3 You are prompted to select the disk on which you want the root to be located. Valid entries are built-in disks d1, d2, and d3, and SMD disks s0, s1, and s2. (These are the only disks that have entries in the restricted mode's CENTIX configuration file.)

   Enter the name of the desired disk and press the RETURN key. Then go to step 4.

Note: *Because the system disk for BTOS already contains valid software, it must not be initialized. Because of space limitations, it is recommended that you not put the root partition on the BTOS system disk.*

If you do select the BTOS system disk, you will not be prompted to initialize it. Go to step 8.

4 You are prompted as to whether you want to initialize the disk you have just selected.

If you do not want to initialize the disk, enter "n" and press the RETURN key. Then go to step 8.

If the disk has never been initialized or you want to reinitialize it, enter "y" and press the RETURN key. Then go to step 5.

5 You are prompted to fill in parameter values for initializing the disk. These parameters correspond to the fields of the BTOS MIVolume utility, which is used to initialize all XE 500 disks.

Refer to the *XE 500 BTOS Operations Reference Manual* for an explanation of the MIVolume parameters.

To set a parameter, enter the desired value in response to the prompt and press the RETURN key. The next parameter prompt is displayed. If the parameter is optional and you want to use the default value, do not enter anything; press the RETURN key to display the next parameter prompt.

6 The last parameter asks you if there are any new bad spots to enter.

If the disk has previously been initialized, enter "n" and press the RETURN key. Then go to step 7.

If the disk has not previously been initialized, enter "y" and press the RETURN key. You are prompted to enter bad spots. Enter the disk's bad spots, pressing the RETURN key after each entry (refer to your *XE 500 CENTIX Administration Guide* for a discussion of disk bad spots.) After entering the last bad spot, press the GO key. Then go to step 7.

7 The parameter list is displayed again. The values you have entered are shown next to the parameter prompts. The system prompts you for whether you want to change any of the parameter values you have entered.

To change a value, enter "y" and press the RETURN key. The entire series of MIVolume parameter prompts is displayed again.

If you are satisfied with the values, enter "n" and press the RETURN key. Then go to step 8.

8 The next series of prompts concerns the size of the root partition.

The system displays the minimum sizes for a CENTIX Run-Time System and a CENTIX Development System root. It then prompts you to supply a size in blocks for the root, with the minimum sizes serving as guidelines (1 block = 512 bytes).

To conserve disk space, you can select the minimum values; however, system performance may be affected. (Increasing the root size provides more space for temporary files and additional software.)

Enter the number of blocks corresponding to the desired size and press the RETURN key.

9 The system prompts you for the partition number for the root partition. Valid partition numbers are 0 through 7. This number is used to identify the BTOS partition file on the disk in which the root file system is to be stored.

Enter a valid partition number and press the RETURN key.

The new root partition is created on the selected disk.

The creation of the root partition includes the following operations:

- The BTOS partition file is created according to the specifications given by the user.
- The CENTIX file systems for the root are made.
- The root partition is mounted in CENTIX.

- A mklost+found operation is performed on the root.
- The root is unmounted.
- The CENTIX configuration file is modified to reflect the partition to which the root is assigned.
- A file system check (**fsck** command) is performed.

After the operations for creating the root have been performed, the system displays a series of prompts for creating the initial swap file for AP00. Refer to the next subsection.

Creating the Swap File

A CENTIX installation requires one swap file for each AP running in the system. Since the initial configuration of a system has only one AP configured into the BTOS master configuration file ([sys]<sys>Master.cnf), the Root/Swap utility only prompts you to create one swap file. The BTOS file name for this file is [*x*]<sys>Swap.Ap.00, where *x* is the volume name (or device name) of the volume on which the swap file is stored.

1 After the root partition has been created, the system prompts you for whether or not you want the swap file on the default disk.

2 To put the swap file on the default disk, enter "y" and press the RETURN key. Then go to step 4.

   To put the swap file on another disk, enter "n" and press the RETURN key. Then go to step 3.

3 You are prompted to select the disk on which you want the swap file to be located. Valid entries are any built-in disks (d1, d2, d3, and so on) or SMD disks (s0, s1, s2, and so on) in the system.

   Enter the name of the desired disk and press the RETURN key. Then go to step 4.

Note: *Because the BTOS system disk already contains valid software, it must not be initialized. If you have selected the BTOS system disk for the swap file, you will not be prompted to initialize it. The swap file is created upon entering the disk name.*

4 You are prompted for whether you want to initialize the disk you have just selected.

If you do not want to initialize the disk, enter "n" and press the RETURN key. Then go to step 8.

If the disk has never been initialized or you want to reinitialize it, enter "y" and press the RETURN key. Then go to step 5.

5 You are prompted to fill in parameter values for initializing the disk. These parameters correspond to the fields of the BTOS MIVolume utility, which is used to initialize all XE 500 disks.

Refer to the *XE 500 BTOS Operations Reference Manual* for an explanation of the MIVolume parameters.

To set a parameter, enter the desired value in response to the prompt and press the RETURN key. The next parameter prompt is displayed. If the parameter is optional and you want to use the default value, do not enter anything; press the RETURN key to display the next parameter prompt.

6 The last parameter asks you if there are any new bad spots to enter.

If the disk has previously been initialized, enter "n" and press the RETURN key. Then go to step 7.

If the disk has not previously been initialized, enter "y" and press the RETURN key. You are prompted to enter bad spots. Enter the disk's bad spots, pressing the RETURN key after each entry (refer to your *XE 500 CENTIX Administration Guide* for a discussion of disk bad spots.) After entering the last bad spot, press the GO key. Then go to step 7.

7 The parameter list is displayed again. The values you have entered are shown next to the parameter prompts. The system prompts you for whether you want to change any of the parameter values you have entered.

   To change a value, enter "y" and press the RETURN key. The entire series of MIVolume parameter prompts is displayed again.

   If you are satisfied with the values, enter "n" and press the RETURN key. Then go to step 8.

8 The system prompts you to enter the size of the swap file. Enter the desired size of the swap file in blocks and press the RETURN key. The minimum size of a swap file is 20000 blocks (10 MB).

   After entering the size and pressing the RETURN key, the swap file is created.

The creation of the swap file includes the following operations:

- The BTOS swap file is created according to the specifications given.
- The CENTIX configuration file is modified to reflect the file to which the swap area is assigned. The swap file is called Swap.Ap.00.

After the operations for creating the swap have been performed, the Setup Software menu of the restricted mode CENTIX is displayed again.

Go now to Section 7 for instructions on how to install the CENTIX software.

Replacing the Existing Root Partition

Once you have already installed your 6.0 release CENTIX system software, you can replace your current root with the standard release root at any time. To do this, use the following procedure.

The procedure assumes that you have booted up in the restricted mode and that you have selected the Set Up Software option from the restricted mode's Main Menu. The restricted mode's Set Up Software Menu should be displayed at the CENTIX terminal.

1 In response to the Set Up Software Menu prompt, enter "2" and press the RETURN key.

The system displays the Root and Swap Location menu, as shown:

```
                ROOT AND SWAP LOCATION

*********************************************
**                                         **
**  0. Exit                                **
**                                         **
**  1. Create an empty root in preparation **
**     for a NEW installation              **
**                                         **
**  2. Move/Resize the root partition      **
**                                         **
**  3. Move/Resize the swap file           **
**                                         **
*********************************************
```

This menu is followed by a prompt for you to choose the desired operation.

2 In response to the Root and Swap Location Menu, enter "1" and press the RETURN key.

The system prompts you to verify that you want to delete the existing root and replace it with a new one.

3 To replace the existing root, enter "y" and press the RETURN key.

The system then prompts you for whether you want to save any customized CENTIX files. A default list of files that you may have customized and would like to save is displayed.

4 Enter either "y" (to save customized files) or "n" (to have the customized files overwritten with new software), then press the RETURN key.

The system then prompts you for any additional files or directories you would like to save.

5 Enter the full CENTIX pathname of each file or directory you want to save, pressing the RETURN key after each entry. If you specify a directory, it is expanded to include all directories and files beneath it.

After the last entry, press the RETURN key twice, then go to step 6.

Note: *The files specified in steps 4 and 5 are saved in a BTOS partition called [sys]<sys>savefiles.centix. Once the save operation is complete, the existing root is deleted.*

The procedure for restoring these files after CENTIX software has been installed and BTOS system files have been configured is provided in the subsection "Restoring Customized CENTIX Files" later in this section.

6 When the file save operation is complete, the system prompts you for whether or not you want the new root partition on the default disk.

To put the root partition on the default disk, enter "y" and press the RETURN key. Then go to step 8.

To put the root partition on another disk, enter "n" and press the RETURN key. Then go to step 7.

7 You are prompted to select the disk on which you want the root to be located. Valid entries are any built-in disks (d1, d2, d3, and so on) or SMD disks (s0, s1, s2, and so on) in the system.

Enter the name of the desired disk name and press the RETURN key. Then go to step 8.

Note: *Because the BTOS system disk already contains valid software, it must not be initialized. Because of space limitations, it is recommended that you not put the root partition on the BTOS system disk.*

If you do select the BTOS system disk, you will not be prompted to initialize it. Go to step 12.

8 You are prompted for whether you want to initialize the disk you have just selected.

If you do not want to initialize the disk, enter "n" and press the RETURN key. Then go to step 12.

If the disk has never been initialized or you want to reinitialize it, enter "y" and press the RETURN key. Then go to step 9.

9 You are prompted to fill in parameter values for initializing the disk. These parameters correspond to the fields of the BTOS MIVolume utility, which is used to initialize all XE 500 disks.

Refer to the *XE 500 BTOS Operations Reference Manual* for an explanation of the MIVolume parameters.

To set a parameter, enter the desired value in response to the prompt and press the RETURN key. The next parameter prompt is displayed. If the parameter is optional and you want to use the default value, do not enter anything; press the RETURN key to display the next parameter prompt.

10 The last parameter asks you if there are any new bad spots to enter.

If the disk has previously been initialized, enter "n" and press the RETURN key. Then go to step 11.

If the disk has not previously been initialized, enter "y" and press the RETURN key. You are prompted to enter bad spots. Enter the disk's bad spots, pressing the RETURN key after each entry (refer to your *XE 500 CENTIX Administration Guide* for a discussion of disk bad spots.) After entering the last bad spot, press the GO key. Then go to step 11.

11 The parameter list is displayed again. The values you have entered are shown next to the parameter prompts. The system prompts you for whether you want to change any of the parameter values you have entered.

To change a value, enter "y" and press the RETURN key. The entire series of MIVolume parameter prompts is displayed again.

If you are satisfied with the values, enter "n" and press the RETURN key. Then go to step 12.

12 The next series of prompts concerns the size of the root partition.

The system displays the minimum sizes for a CENTIX Run-Time System and a CENTIX Development System root. It then prompts you to supply a size in blocks for the root, with the minimum sizes serving as guidelines (1 block = 512 bytes).

To conserve disk space, you can select the minimum values; however, system performance may be affected. (Increasing the root size provides more space for temporary files and additional software.)

Enter the number of blocks corresponding to the desired size and press the RETURN key.

13 The system prompts you for the partition number for the root partition. Valid partition numbers are 0 through 7. This number is used to identify the BTOS partition file on the disk in which the root file system is to be stored.

Enter a valid partition number and press the RETURN key.

A series of messages is displayed, indicating that the new root partition is being created on the selected disk.

The creation of the root partition includes the following operations:

- The BTOS partition file is created according to the specifications given by the user.
- The CENTIX file systems for the root are made.
- The root partition is mounted in CENTIX.
- A mklost+found operation is performed on the root.
- The root is unmounted.
- The CENTIX configuration file is modified to reflect the partition to which root is assigned.
- A file system check (**fsck** command) is performed.

When the root partition has been created, you are returned to the Root and Swap Location Menu.

Changing a Root Partition

To move or change the size of an existing root partition, use the following procedure. The procedure assumes that you have booted up in the restricted mode and that the restricted mode's Set Up Software Menu is displayed at the terminal.

Note: *If you want to resize the current root but not move it, you cannot use the following procedure. Instead, use the procedure described under "Replacing the Existing Root Partition."*

If you want to move the current root to another partition on the same disk and there is not enough disk space to store both partitions, you cannot use the following procedure. Instead, use the procedure described under "Replacing the Existing Root Partition."

1 In response to the Set Up Software Menu prompt, enter "2" and press the RETURN key.

The system displays the Root and Swap Location menu, as shown:

```
                ROOT AND SWAP LOCATION

**********************************************
**                                          **
** 0. Exit                                  **
**                                          **
** 1. Create an empty root in preparation   **
**    for a NEW installation                **
**                                          **
** 2. Move/Resize the root partition        **
**                                          **
** 3. Move/Resize the swap file             **
**                                          **
**********************************************
```

This menu is followed by a prompt for you to choose the desired operation.

2 In response to the Root and Swap Location Menu prompt, enter "2" and press the RETURN key.

The system displays the location of the existing root partition. It then prompts you for whether or not you want to move the root partition.

3 To move the root partition, enter "y" and press the RETURN key.

4 The system prompts you for whether or not you want the root partition on the default disk.

To put the root partition on the default disk, enter "y" and press the RETURN key. Then go to step 6.

To put the root partition on another disk, enter "n" and press the RETURN key. Then go to step 5.

5 You are prompted to select the disk on which you want the root to be located. Valid entries are any built-in disks (d1, d2, d3, and so on) or SMD disks (s0, s1, s2, and so on) in the system.

Enter the name of the desired disk and press the RETURN key. Then go to step 6.

Note: *Because the BTOS system disk already contains valid software, it must not be initialized. Because of space limitations, it is recommended that you not put the root partition on the BTOS system disk.*

If you do select the BTOS system disk, you will not be prompted to initialize it. Go to step 10.

6 You are prompted for whether you want to initialize the disk you have just selected.

If you do not want to initialize the disk, enter "n" and press the RETURN key. Then go to step 10.

If the disk has never been initialized or you want to reinitialize it, enter "y" and press the RETURN key. Then go to step 7.

7 You are prompted to fill in parameter values for initializing the disk. These parameters correspond to the fields of the BTOS MIVolume utility, which is used to initialize all XE 500 disks.

Refer to the *XE 500 BTOS Operations Reference Manual* for an explanation of the MIVolume parameters.

To set a parameter, enter the desired value in response to the prompt and press the RETURN key. The next parameter prompt is displayed. If the parameter is optional and you want to use the default value, do not enter anything; press the RETURN key to display the next parameter prompt.

8 The last parameter asks you if there are any new bad spots to enter.

If the disk has previously been initialized, enter "n" and press the RETURN key. Then go to step 9.

If the disk has not previously been initialized, enter "y" and press the RETURN key. You are prompted to enter bad spots. Enter the disk's bad spots, pressing the RETURN key after each entry (refer to your *XE 500 CENTIX Administration Guide* for a discussion of disk bad spots.) After entering the last bad spot, press the GO key. Then go to step 9.

9 The parameter list is displayed again. The values you have entered are shown next to the parameter prompts. The system prompts you for whether you want to change any of the parameter values you have entered.

To change a value, enter "y" and press the RETURN key. The entire series of MIVolume parameter prompts is displayed again.

If you are satisfied with the values, enter "n" and press the RETURN key. Then go to step 10.

10 The system prompts you for the partition number for the root partition. Valid partition numbers are 0 through 7. This number is used to identify the BTOS partition file on the disk in which the root is to be stored.

   Enter a valid partition number and press the RETURN key.

11 The system displays the size of the existing root partition in blocks. It then prompts you for whether or not you want to change the size of the root partition.

   To change the size of the root partition, enter "y" and press the RETURN key. Then go to step 12.

   To leave the root partition at its current size, enter "n" and press the RETURN key. The root partition is moved.

12 The system displays the minimum sizes for a CENTIX Run-Time System and a CENTIX Development System root. It then prompts you to supply a size in blocks for the root, with the minimum sizes serving as guidelines (1 block = 512 bytes).

   To conserve disk space, you can select the minimum values; however, system performance may be affected. (Increasing the root size provides more space for temporary files and additional software.)

   Enter the number of blocks corresponding to the desired size and press the RETURN key.

   The root partition is moved and resized.

The moving and/or resizing of the root partition includes the following operations:

- The BTOS partition file for the root is created according to the specifications given.

  If a new size was specified, the partition is created with the new size. Otherwise, the size of the root remains the same.

- The CENTIX configuration file is modified to reflect the disk and partition to which the root has been moved.

After the operations for moving and/or resizing the root have been performed, the Root and Swap Location Menu is displayed again.

Changing a Swap File

To move or change the size of an existing swap file, use the following procedure. The procedure assumes that you have booted up in the restricted mode and that the restricted mode's Set Up Software Menu is displayed at the terminal.

1 In response to the Set Up Software Menu prompt, enter "2" and press the RETURN key.

The system displays the Root and Swap Location menu, as shown:

```
                ROOT AND SWAP LOCATION

.............................................
..                                         ..
** 0. Exit                                 **
..                                         ..
** 1. Create an empty root in preparation  **
**    for a NEW installation               **
..                                         ..
** 2. Move/Resize the root partition       **
..                                         ..
** 3. Move/Resize the swap file            **
..                                         ..
.............................................
```

This menu is followed by a prompt for you to choose the desired operation.

2 In response to the Root and Swap Location Menu prompt, enter "3" and press the RETURN key.

The system displays the location of the existing swap file associated with AP00. It then prompts you for whether or not you want to move the swap file.

3 To move the swap file, enter "y" and press the RETURN key. Then go to step 5.

To leave the swap file at its current location, enter "n" and press the RETURN key. Then go to step 10.

4 The system prompts you for whether or not you want the swap file on the default disk.

To put the swap file on the default disk, enter "y" and press the RETURN key. Then go to step 6.

To put the swap file on another disk, enter "n" and press the RETURN key. Then go to step 5.

5 You are prompted to select the disk on which you want the swap to be located. Valid entries are any built-in disks (d1, d2, d3, and so on) or SMD disks (s0, s1, s2, and so on) in the system.

Enter the name of the desired disk and press the RETURN key. Then go to step 6.

Note: *Because the BTOS system disk already contains valid software, it must not be initialized. If you have selected the BTOS system disk for the swap file, you will not be prompted to initialize it. Go to step 10.*

6 You are prompted for whether you want to initialize the disk you have just selected.

If you do not want to initialize the disk, enter "n" and press the RETURN key. Then go to step 10.

If the disk has never been initialized or you want to reinitialize it, enter "y" and press the RETURN key. Then go to step 7.

7 You are prompted to fill in parameter values for initializing the disk. These parameters correspond to the fields of the BTOS MIVolume utility, which is used to initialize all XE 500 disks.

Refer to the *XE 500 BTOS Operations Reference Manual* for an explanation of the MIVolume parameters.

To set a parameter, enter the desired value in response to the prompt and press the RETURN key. The next parameter prompt is displayed. If the parameter is optional and you want to use the default value, do not enter anything; press the RETURN key to display the next parameter prompt.

8 The last parameter asks you if there are any new bad spots to enter.

If the disk has previously been initialized, enter "n" and press the RETURN key. Then go to step 9.

If the disk has not previously been initialized, enter "y" and press the RETURN key. You are prompted to enter bad spots. Enter the disk's bad spots, pressing the RETURN key after each entry (refer to your *XE 500 CENTIX Administration Guide* for a discussion of disk bad spots.) After entering the last bad spot, press the GO key. Then go to step 9.

9 The parameter list is displayed again. The values you have entered are shown next to the parameter prompts. The system prompts you for whether you want to change any of the parameter values you have entered.

To change a value, enter "y" and press the RETURN key. The entire series of MIVolume parameter prompts is displayed again.

If you are satisfied with the values, enter "n" and press the RETURN key. Then go to step 10.

10 The system displays the size of the existing swap file in blocks. It then prompts you for whether or not you want to change the size of the swap file.

To change the size of the swap file, enter "y" and press the RETURN key. Then go to step 11.

To leave the swap file at its current size, enter "n" and press the RETURN key. The swap file is moved to the disk that you have selected. If there is another swap file in the system, the utility cycles again through the series of prompts described in steps 3 through 10.

11 Enter the desired size of the swap file in blocks and press the RETURN key. The minimum size of a swap file is 20000 blocks (10 MB).

After entering the size and pressing the RETURN key, the swap file is moved to the disk that you have selected and is resized. If there is another swap file in the system, the utility cycles again through the series of prompts described in steps 3 through 10.

The moving and/or resizing the swap file includes the following operations:

- The new swap file is created according to the specifications given by the user. If a new size was specified, the file is created with the new size.
- The CENTIX configuration file is modified to reflect the disk and partition file to which the swap file has been moved.

After the operations for moving and/or resizing the swap area have been performed, the Root and Swap Location Menu is displayed again. You can return to the Set Up Software Menu again by selecting option 0 (exit the Root/Swap utility).

Restoring Customized CENTIX Files

To restore the customized CENTIX files that you saved when running the Root/Swap utility, enter the following command from the CENTIX shell:

```
$ /etc/.restore
```

This command causes all of the files listed in the /etc/savefile file to be copied back into the CENTIX file systems. Corresponding 6.0 release files are overwritten by the customized files.

This procedure is also used to restore customized files that are saved during the installation of other CENTIX products. Refer to Section 7 for a description of the CentixLoad utility install scripts.

Installing XE 500 CENTIX Software

Once the root partition and AP00 swap file have been created, you can install the XE 500 CENTIX system software into the root. This is done at the CENTIX terminal through the interactive CentixLoad utility.

The CentixLoad utility provides a menu-driven facility through which you can

- Install all of the standard CENTIX system software from the release media.
- Install all mandatory products but selectively install optional products of the CENTIX standard system software from the release media. This capability is only available when the release media are disk cartridges.

The CentixLoad utility can also be used to reinstall 6.0 release CENTIX system software.

The CentixLoad utility allows you to change release media in the middle of a session. Therefore, you can load software from both tape and disk cartridge media during the same session.

During a CentixLoad session, error handling and recovery is performed interactively through error menus. The error menus allow you to select the type of action to take to recover from an error condition. A log file, /CentixLoad.log, is also kept of the user interaction and operations that take place during a session. This log file is kept in the restricted mode CENTIX root file system. Refer to the subsection "Diagnosing CentixLoad Utility Problems" in Section 11.

Note: *After completing a CentixLoad utility session, you should always print the log file. The log file is automatically removed the next time the system is booted in restricted mode.*

This section describes how to run the CentixLoad utility, how to use it to install CENTIX system software, and how it installs CENTIX products.

Caution: *The CentixLoad utility can also be used to install certain CENTIX application software products. However, when CentixLoad is invoked while in the restricted mode, it should only be used to install CENTIX system software products.Refer to Section 9 for instructions on installing CENTIX application software products.*

Installing the System Software

Caution: *The system must always be in the restricted mode to install CENTIX system software.*

Perform the following procedure to install CENTIX system software.

1 If you have just completed creating the swap partition, as described in Section 6, the restricted mode's Set Up Software Menu and prompt are displayed at the terminal.

  At the Set Up Software Menu prompt, enter "1" and press the RETURN key. This causes the CentixLoad utility to begin. The CentixLoad Main Menu is displayed.

Note: *If you need detailed information about using the CentixLoad utility, refer to the next subsection, "Using the CentixLoad Utility."*

2 Load the CENTIX run-time system medium of your CENTIX software release set.

  For 8-user systems using disk cartridges, load the first disk, marked "BN8-D1." For 8-user systems using tapes, load the tape marked "BN8-T" or "BN8-C."

  For 16- and 32-user systems using disk cartridges, load the first disk, marked "BNX-D1." For 16- and 32-user systems using tapes, load the tape marked "BNX-T" or "BNX-C."

3 At the Main Menu prompt, enter the number corresponding to the medium type and press the RETURN key.

The Insert Medium Menu and prompt are displayed.

4 At the Medium Menu prompt, enter "1" and press the RETURN key.

If you are installing CENTIX software from tape, all of the CENTIX products on the tape are installed into the XE 500 CENTIX system. When the installation operation is complete, you are returned to the CentixLoad Main Menu. Go to step 7.

If you are installing software from a disk cartridge, the Loading Method Menu is displayed. The Loading Method Menu allows you to install all products on the disk cartridge automatically or to skip installing optional CENTIX software. Go to step 5.

5 At the Loading Method Menu prompt, enter the number corresponding to the loading method you want to use and press the RETURN key.

If you selected to load all software automatically, the utility displays a message indicating when the installation operations are complete. You are returned to the Main Menu.

If you are selectively loading software, the utility first installs all CENTIX software on the disk cartridge required to run the system. It then displays the Product Load Menu for each product not required to run the system. The Product Load Menu allows you to choose to install the optional software product.

When the utility has cycled through all products on the release disk cartridge, it displays a message indicating that the installation operations are complete. You are returned to the Main Menu.

6 Repeat steps 2 through 5 for the remaining two disk cartridges.

For 8-user systems, load the disk marked "BN8-D2," then load the last disk, marked "BN8-D3."

For 16- and 32-user systems, load the disk marked "BNX-D2," then load the last disk, marked "BNX-D3."

When you have processed the last disk, go to step 7.

7 If you are not installing CENTIX Development System software, go to step 8.

If you are installing CENTIX Development System software, repeat steps 2 through 5 for the Development System software release media.

For systems using disk cartridges, load the first disk, marked "BND-D1," then load the second disk, marked "BND-D2."

For systems using tapes, load the tape marked "BND-T" or "BND-C."

When you have completed installing Development System software, go to step 8.

8 At the CentixLoad Main Menu prompt, enter "0" and press the RETURN key.

You are returned to the restricted mode's Set Up Software Menu.

9 Go to Section 14 for instructions on how to configure the BTOS system configuration files. Sections 12 and 13 provide background information about the BTOS system configuration files.

Using the CentixLoad Utility

The CentixLoad utility is a menu-driven facility through which you install CENTIX software products from release media.

Each CentixLoad menu comprises

- A heading that describes the purpose of the menu.
- A list of actions from which to select. Each action is assigned a number, which is used to enter your selection.
- A prompt at which you enter the number of the action you want to execute. After entering the desired number, you must press the RETURN key to execute the action. If you type an invalid entry, the terminal beeps and the invalid entry is not displayed.

The Main Menu provides the option of leaving the utility and returning to the restricted mode's Set Up Software Menu. All other menus provide the option of returning to the Main Menu. Therefore, you are always no more than one menu away from being able to terminate the utility.

Main Menu

When the utility is invoked, the Main Menu is displayed. The Main Menu gives you the option to

- Exit the utility.
- Select the type of medium from which you want to install software: disk cartridge, half-inch tape, or quarter-inch cartridge (QIC) tape.

The Main Menu and prompt are displayed as follows:

```
                    MAIN MENU

**************************************************
**                                              **
**  0. Exit CentixLoad                          **
**                                              **
**  1. Disk Cartridge                           **
**                                              **
**  2. Half-Inch Tape                           **
**                                              **
**  3. Quarter-Inch Cartridge (QIC) Tape        **
**                                              **
**************************************************

Please Select Medium ==>
```

To return to the process from which you invoked the utility, enter "0" and press the RETURN key.

To install software, enter the number corresponding to the appropriate medium type and press the RETURN key. The Insert Medium Menu is displayed.

Insert Medium Menu

After selecting the medium type, the Insert Medium Menu and prompt are displayed as follows:

```
                    INSERT MEDIUM MENU

..................................................
..                                              ..
..          0. Return to Main Menu              ..
..                                              ..
..          1. Medium mounted and on line       ..
..                                              ..
..................................................

Enter Selection ==>
```

Insert the CENTIX software release medium. If the medium is a tape, wait for it to come on-line. If the medium is a disk cartridge, wait for it to come up to speed.

Note: *It does not matter in what order you install the CENTIX software release media.*

When the medium has been inserted and is on-line or up to speed, enter "1" in response to the Insert Medium Menu prompt and press the RETURN key.

If you are installing software from a tape, all of the CENTIX products on the tape are installed into the XE 500 CENTIX system. When the installation operation is complete, you are returned to the CentixLoad Main Menu. From the Main Menu you can either exit the utility or install more CENTIX software.

If you are installing software from a disk cartridge, the Loading Method Menu is displayed.

Loading Method Menu

The Loading Method Menu menu allows you to

- Return to the Main Menu.
- Automatically install all of the software products on the disk cartridge.
- Selectively install optional software products from the disk cartridge.

The Loading Method Menu and prompt are displayed as follows:

```
                    LOADING METHOD MENU

**************************************************
**                                              **
**  0. Return to Main Menu                      **
**                                              **
**  1. Load all Products on this Cartridge      **
**     Automatically                            **
**                                              **
**  2. Load Products on this Cartridge Selectively **
**                                              **
**************************************************

Enter Selection ==>
```

To load all system software from the disk cartridge, enter "1" and press the RETURN key. CentixLoad begins installing products. As products are installed, the utility displays messages that indicate the various installation operations being performed. When all products on the disk cartridge have been installed, you are returned to the Main Menu. From the Main Menu you can either exit the utility or install more CENTIX software.

To selectively load optional products on the disk cartridge, enter "2" and press the RETURN key. CentixLoad first installs all mandatory products stored on the disk cartridge. It then searches for the first optional product and displays the Product Load Menu.

Product Load Menu

The Product Load Menu allows you to

- Return to the Main Menu.
- Install an optional CENTIX software product.
- Not install an optional CENTIX software product and skip to the next product.

The Product Load Menu is displayed as follows:

```
                PRODUCT LOAD MENU

..................................................
..                                              ..
..          0. Go on to Next Medium             ..
..                                              ..
..          1. Load this Product                ..
..                                              ..
..          2. Skip this Product                ..
..                                              ..
..................................................
```

Following the Product Load Menu is the product name and, depending on the product, a descriptive comment about the product. This information is followed by the prompt

```
Enter Selection ==>
```

If you do not want to load any other products from the disk cartridge, enter "0" and press the RETURN key. You are returned to the Main Menu.

To load the product being displayed, enter "1" and press the RETURN key. The product is installed into the XE 500 CENTIX system and the utility searches for the next product. If another product is found, the Product Load Menu is displayed again.

To skip loading the product being displayed and go on to the next one, enter "2" and press the RETURN key. The utility searches for the next product. If another product is found, the Product Load Menu is displayed again.

When all of the products on the disk cartridge have been processed, you are returned to the Main Menu. From the Main Menu you can either exit the utility or install more CENTIX software.

How the CentixLoad Utility Works

Installing Disk Cartridge-Based Products

To install a disk cartridge-based product, CentixLoad first renames the product to [d0]<sys>Partition.7. It then performs a file system check on the partition using the **fsck** command and mounts it as /.install. CentixLoad creates the directory /.install itself and removes it upon completion.

After mounting the partition, CentixLoad runs a script file called "install" within the partition. This script loads the contents of the partition into the correct place in the CENTIX file system. It may ask for user inputs and/or create its own file system. The install script may also modify CENTIX configuration files such as /etc/passwd or create new BTOS files.

Once the install script completes, CentixLoad unmounts the partition and renames it back to its original name. It also removes the mount point /.install.

Since it is possible for fatal errors to occur while loading a cartridge, (that is, the user may kill CentixLoad or the cartridge may be removed before the rename occurs), CentixLoad first checks a cartridge for the existence of [d0]<sys>Partition.7. If this file is present, CentixLoad examines the cartridge to determine if any files are missing. If just one file is missing, CentixLoad will rename Partition.7 to the missing file. If no files or more than one file are missing, CentixLoad cannot load any of the products on the cartridge and returns an error message to that effect. Additional missing files or an extra Partition.7 file can only occur if the user has modified the cartridge or if the cartridge has been damaged.

Installing Tape-Based Products

All tapes loaded by CentixLoad are created in the **cpio -cB** command format. CentixLoad first loads the file "install" into the root directory (/) and runs it. This file is a shell script, similar to the script that appears within partitions, that installs the product from the tape. After the product is installed, CentixLoad removes the install script.

Install Scripts

On each CENTIX software release medium, an install script causes all of the files found on the medium to be copied into the CENTIX root. This is true for both CENTIX system software and CENTIX application software releases.

A file system check is performed on the root before mounting it. If this is the first release medium of the software product, the system runs through the following procedure. This procedure allows you to save customized CENTIX files

1 The system prompts you for whether you want to save any related customized CENTIX files. A default list of files is displayed.

  Enter either "y" (to save customized files) or "n" (to have the customized files overwritten with new software), then press the RETURN key.

2 The system then prompts you for any additional files or directories you would like to save.

  Enter the full CENTIX path name of each file or directory you want to save, pressing the RETURN key after each entry. If you specify a directory, it is expanded to include all directories and files beneath it.

  After the last entry, press the RETURN key twice.

The files specified are saved in a BTOS partition called [sys]<sys>savefiles.centix. Once the save operation is complete, the existing root is deleted.

The procedure for restoring these customized files after the CENTIX software has been installed is provided in Section 6.

A message is then displayed telling you that the copy operation for the software installation has begun. The system displays a list of the files as they are being copied. When the copy is finished, the system displays a message to that effect.

Depending on the product being installed, the install script on the first release medium may also

- Execute the run file /SAF/install1.
- Create the file [sys]<sys>Centix.sys
- Create devices in the /dev directory.

Finally, the root partition is unmounted and a file system check is performed on the root.

Supporting a Mixed Environment

This section includes the procedure for installing BTOS and CENTIX system software so that the XE 500 can operate in a BTOS and CENTIX environment.

In the BTOS environment, the XE 500 is a shared resource processor for the BTOS workstations connected to it. In the CENTIX environment, the XE 500 is the CENTIX processor for CENTIX terminals.

Abbreviated Outline of the Installation Procedure

This manual and the *XE 500 BTOS Installation and Implementation Guide* must be used in conjunction to carry out the installation procedure. Some of the steps must be executed at the CENTIX terminal, others from a BTOS workstation.

The following outline lists the steps of the installation procedure. Included are references to where these steps are described and at which device, a CENTIX terminal or a workstation, the steps are performed.

This outline provides only an abbreviated version of the actual procedure. You should be familiar with the complete procedures described in both manuals before attempting to install software.

STEP 1: Install the Restricted Mode Software from the Boot Load Release Medium

Execute this step from the CENTIX terminal. Refer to the subsection "Executing the Boot Load Utility" in Section 4 of this manual.

STEP 2: Install the BTOS Workstation Software from the Workstation Release Floppy Disks

Execute this step from the workstation. Refer to Section 5 of the *XE 500 BTOS Installation and Implementation Guide*.

STEP 3: Install the "BSW" BTOS System Software

This is the BTOS system software that supports the workstations. Execute this step from the workstation. Refer to the subsection "Installing the System Software" in Section 6 of the *XE 500 BTOS Installation and Implementation Guide*.

STEP 4: Install the "BTOS" BTOS System Software

This is the BTOS system software that supports CENTIX. Execute this step from the CENTIX terminal. Refer to the subsection "Installing the BTOS System Software" in Section 5 of this manual.

STEP 5: Create the Root Partition

Execute this step from the CENTIX terminal. Refer to the subsection "Creating the Root Partition" in Section 6 of this manual.

STEP 6: Create the Swap File for AP00

Execute this step from the CENTIX terminal. Refer to the subsection "Creating the Swap File" in Section 6 of this manual.

STEP 7: Install the CENTIX System Software from the CENTIX Release Media

Execute this step from the CENTIX terminal. Refer to the subsection "Installing the CENTIX System Software" in Section 7 of this manual.

STEP 8: Create the BTOS System Configuration Files

Execute this step from the CENTIX terminal or the workstation. Refer to the subsection "Creating the BTOS System Configuration Files" in Section 14 of this manual.

STEP 9: Reboot the System in Normal Mode

Refer to the subsection "Rebooting the System after Running MBTOS Config" in Section 14 of this manual.

STEP 10: Install CENTIX Environmental Software, Configure CENTIX, and Perform Post-Installation Tasks

Execute this step from the CENTIX terminal. Section 9 covers installing CENTIX environmental software. Section 15 outlines configuring the CENTIX system. Section 16 describes post-configuration tasks.

STEP 11: Add the Temporary Directory Filter to the Master Processor Initialization File

Execute this step from the CENTIX terminal or the workstation. Use the MBTOS Config utility to add the Temporary Directory Filter service to the master processor initialization file. Refer to Section 14 of this manual.

STEP 12: Create the BTOS Command Files

Execute this step from the CENTIX terminal or the workstation. Refer to the subsection "BTOS Command Forms and User Command Files" in Section 12 of the *XE 500 BTOS Installation and Implementation Guide*.

STEP 13: Customize BTOS System Services

Execute this step from the CENTIX terminal or the workstation. Refer to Sections 13 through 16 of the *XE 500 BTOS Installation and Implementation Guide.*

Installing CENTIX Environmental Software

There are two packaging formats for release media that contain CENTIX-based environmental software products (the term *environmental software products* refers to software such as the CENTIX programming languages, ISAM, centreSCREEN, and so on).

You must use the Install CENTIX Products facility of centrEASE to install software from media using either packaging format. The difference is that

For one format, centrEASE calls up the MSysLoad utility to install the release software onto a temporary partition on the BTOS system disk. The software is then copied from the temporary partition into the CENTIX file systems. This format is used for most application products available prior to the 5.0 level release of the CENTIX system software.

For the other format, centrEASE calls the CentixLoad utility to install the software directly from the release medium into the CENTIX file systems. Install scripts cause any other system operations that are necessary to install the software to be run. This format is new with the 5.0 release of the CENTIX system software.

In either case, refer to the *XE 500 CENTIX centrEASE Operations Reference Manual* for the procedure for running the Install CENTIX Products facility of centrEASE.

Cautions:

1 *Do not use the keyswitch for loading pre-5.0 release environmental software products. centrEASE now does all of the operations that were previously controlled by keyswitch positions.*

2 *If you are reinstalling a product from old release media, be careful not to install any old system software that would overwrite current release system software.*

3 *For pre-5.0 environmental software products on release disk cartridges, do not use the SysLoad cartridge.*

The application software product's documentation should describe any pre- or post-installation steps that must be performed as part of the application's installation and configuration.

Note: *Some application software products may require special BTOS system services. The run statement for one of these services is automatically added to the appropriate processor initialization file during the installation.*

If you want to move such a service's run statement to another processor initialization file, it cannot be done through the MBtos Config utility. The appropriate initialization files must be modified manually using an editor.

BootLoadParams File

The BootLoadParams file is a BTOS file that contains parameters to be used when the system disk is initialized during the Boot Load utility.

The presence of the BootLoadParams file allows you to tailor the MIVolume parameters for your system disk. Once the Boot Load utility has been run, the BootLoadParams file, [sys]<sys>BootLoadParams, is stored on the BTOS system disk.

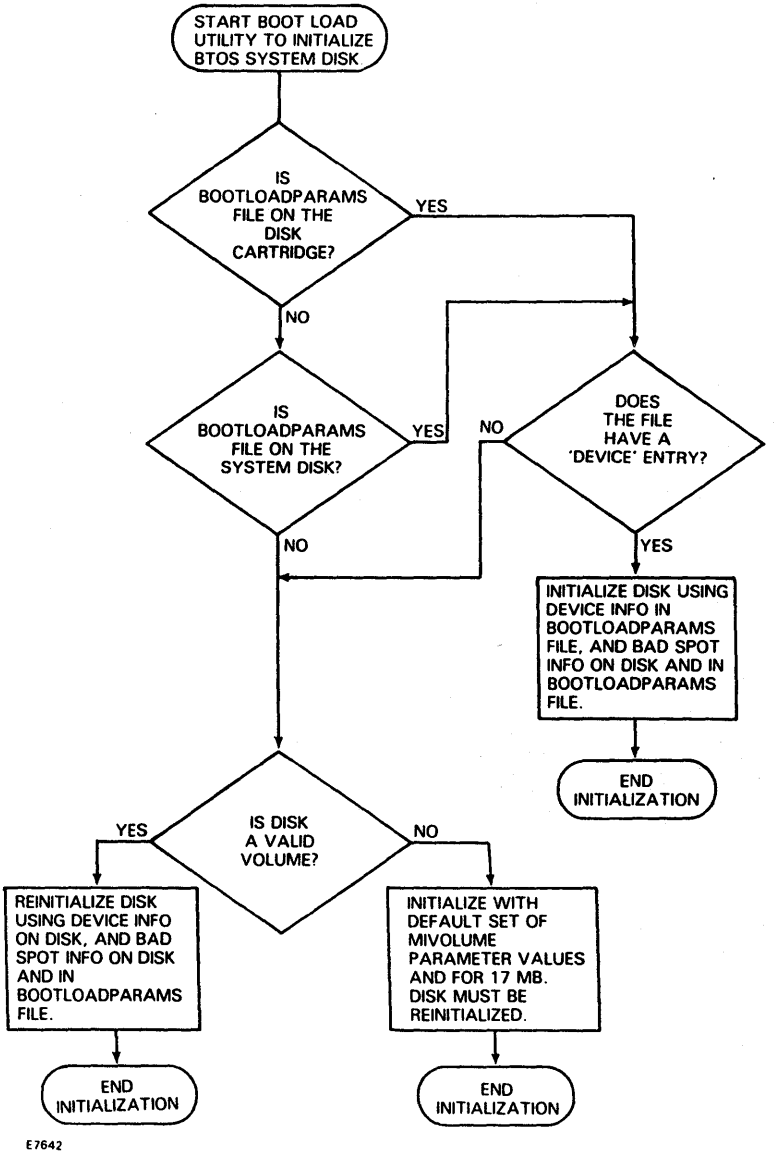
How the BootLoadParams File Is Used

Figure 10-1 is a flow chart that outlines the system disk initialization process during the Boot Load utility.

If a system disk is invalid or not initialized, and no BootLoadParams file exists, there is no source for device and bad spot information. The device is formatted for 17 MB and no bad spots are recorded. Default values, discussed in the next subsection, are used for all other MIVolume utility parameters. The procedure for checking if the system disk is improperly formatted is in Section 5. Section 11 includes the recovery procedure for an improperly formatted system disk.

If you run the Boot Load utility using a Boot Load disk cartridge, the system first looks on the disk cartridge for the BootLoadParams file. A BootLoadParams file is not included on the Boot Load disk cartridge. However, you can copy a BootLoadParams file to the <sys> directory of the disk cartridge or create one with the **ofed** or **ofvi** editor. You can then modify the BootLoadParams file on the disk cartridge to include a "Device" entry for your system disk and a listing of its bad spots. This will allow you to properly initialize an invalid system disk through the Boot Load utility.

Figure 10-1 **Boot Load System Disk Initialization Process**



Modifying BootLoadParam MIVolume Parameter Values

The default BootLoadParams file for an XE 500 system with no SMDs in the base enclosure is shown below:

```
:01: MIVolume !d1:
device,d1
device password,d1
volume name
volume passwd
sysimage size
logFile
CrashFile
MaxDirectories
MaxFiles
PrimHeaders
MaxFilesInSys
SysDirPasswd
WriteProtSys
SuppressFormat
SurfaceTest
Debug
LogFile
Device
BadSpots
```

The default BootLoadParams file for an XE 500 system with SMDs in the base enclosure is shown below:

```
:01: MIVolume !s0:
device,s0
device password,s0
volume name
volume passwd
sysimage size
logFile
CrashFile
MaxDirectories
MaxFiles
PrimHeaders
MaxFilesInSys
SysDirPasswd
WriteProtSys
SuppressFormat
SurfaceTest
Debug
LogFile
Device
BadSpots
```

The lack of a value for a parameter in the BootLoadParams file means that, when this file is being used by the Boot Load utility, the default value for that parameter is to be used. Table 10-1 lists the built-in BootLoadParams default values.

Table 10-1 **BootLoadParams Default MIVolume Parameters**

Device	d1 or s0
Device password	d1 or s0
Volume name	
Volume password	
Size of SysImage.sys	470
Size of system log file	32
Size of crash file	1600
Maximum directories on volume	500
Maximum files in volume	2000
Primary headers only?	no
Maximum files in Sys directory	1000
Password for Sys directory	
Write protect Sys directory	no
Suppress format?	no
Number of surface tests?	1 for d1, 3 for s0
Debug	no
Log file	
Device type	
Bad spots	(Boot Load reads the system disks volume structure files for bad spot information.)

You can reinitialize the system disk using the BootLoadParams file as the input for the BTOS MIVolume utility parameters but with nondefault values. To do this,

1 Using the ofed or ofvi editor, enter a comma after the appropriate parameter name.

2 Enter the appropriate value. Refer to the description of the **MIVolume** command form parameters in the *XE 500 BTOS Operations Reference Manual.*

   If you want to enter a null value for the parameter, enter only a comma after the parameter name.

   If you want to use the default value for a parameter, do not add anything after the parameter name.

3 Repeat steps 1 and 2 for each parameter that you want to specify.

Caution: *When entering bad spots, you must enter the designation for each known bad spot. Use only spaces between bad spot designations—do not use carriage returns.*

---

4 When you have finished modifying the file, exit the editor.

5 Shutdown the CENTIX system.

6 Put the system's Boot Load medium on line.

7 Turn the keyswitch at the front panel to STOP and then to REMOTE.

   This causes the system disk to be reinitialized with MIVolume parameters as specified in the [sys]<sys>BootLoadParams file.

Any syntax error in the BootLoadParams file causes the STATUS display to hang at 82 when the Boot Load utility is executed. If this occurs, correct the file and run the Boot Load utility again.

Notes :

1 *For Volume Name, Boot Load checks the disk for an existing valid volume name and uses it. If a valid volume name is not found on disk, the volume name "disk1" is used.*

2 *For the Device Type, Boot Load determines the current device type. If there is not one, the Device type field says "unknown" and the disk is formatted for only about 17 MB (enough disk space to be able to install system software). Refer to the procedure in Section 11 for recovering from an improperly initialized system disk.*

When modifying the BootLoadParams file, observe the following guidelines:

- The order of the entries must appear in the order shown for the default file. The Boot Load utility reads this file sequentially, taking the first parameter specified to be the device, the second parameter to the device password, the 18th parameter is to be device type, and so on.
- Each parameter to be input to the Boot Load utility appears in the format *description, parameter*. Text up to the comma is considered comment text. The comma must appear in order for the parameter to input into the Boot Load utility.

  For example, using the default file as it is would maintain the current volume password. Adding only a comma after "volume password" would remove the current volume password and leave the volume unprotected.
- The bad spot entry in the BootLoadParams file should list the known bad spots for the system disk. A bad spot entry can continue until an end of file. The bad spot information contains as many bad spots that can fit into a 1024 byte array, or approximately 60 bad spots. A comma must appear after the bad spot label in order for any bad spots to be input into the Boot Load utility. All bad spots must be delimited by a single space, not a carriage return.

As an example, the following modified BootLoadParams file would cause the system disk, d1, to be initialized as a 71.3-MB disk with the bad spots as indicated. The default BootLoadParams parameter values would be used for all other MIVolume parameters.

```
:01: MIVolume !d1:
device,d1
device password,d1
volume name
volume passwd
sysimage size
logFile
CrashFile
MaxDirectories
MaxFiles
PrimHeaders
MaxFilesInSys
SysDirPasswd
WriteProtSys
SuppressFormat
SurfaceTest
Debug
LogFile
Device,Micropolis86
BadSpots,3/0/2418/10 61/0/251/10 540/7/2241/10
3/1/2418/10 61/1/251/10 540/6/2241/10 3/2/2418/10
61/2/251/10 540/5/2241/10
```

Troubleshooting Software Installation Problems

This section provides information and procedures that will help you to troubleshoot software installation problems. There is a subsection for each of the utilities that are used during the installation process. There is also a subsection that discusses how to reinitialize a BTOS system disk that had been uninitialized prior to running the Boot Load utility.

Your main tools for troubleshooting problems are the logs generated by the utilities. These logs contain a record of the system operations, and any errors, that occur during a session of the utility. The logs are:

Boot Load utility log. This log, which is the BTOS file [sys]<sys>BootLoad.log, contains a record of the Boot Load session. The record of each session is appended to this file. This log can be examined while in the restricted mode or normal mode.

MSysLoad utility log. This log, which is the BTOS file [sys]<sys>SysLoad.log, contains a record of the MSysLoad session. The record of each session is appended to this file. During an MSysLoad session, BTOS software is installed. This log can be examined while in the restricted mode or normal mode.

Root/Swap utility log. This log, which is the CENTIX file /rootswap.log, contains a record of a Root/Swap session. The previous session's log file is removed at the start of a new session. Because this file is stored in the restricted mode root, it can only be examined while in the restricted mode.

CentixLoad utility log. This log, which is the CENTIX file /CentixLoad.log, contains a record of a CentixLoad session. The previous session's log file is removed at the start of a new session. Because this file is stored in the restricted mode root, it can only be examined while in the restricted mode.

The MBTOS Config utility does not keep a log file. However, the previous versions of any files modified through this utility are saved.

All centrEASE sessions are recorded in the centrEASE log file. Refer to the *XE 500 CENTIX centrEASE Reference Manual* for more information about this log.

Diagnosing Boot Load Installation Problems

Normal Operations—Disk Cartridge

The Boot Load utility uses the front panel STATUS display to indicate the progress of software installation. Table 11-1 lists the codes displayed and the corresponding Boot Load function for running Boot Load from a disk cartridge.

Table 11-1 **Boot Load Codes—Normal Operation with Disk Cartridge**

Code	Meaning
80	Start Boot Load utility and create BootLoad.log file.
81	Pause for MIVolume. User must turn the keyswitch to NORMAL to initialize the system disk.
82	Initializing the system disk.
86	Creating the directories listed in ProductContent.
87	Copying the system software from the release disk cartridge to the system disk.
88	Appending text to the appropriate file as listed in the ProductContent file.
89	Creating the files specified in ProductContent.
90	Successful completion of the software installation from the currently loaded release disk cartridge.

Interpreting Error Conditions—Disk Cartridge

A status code from 91 to 99 indicates that an error has occurred. Table 11-2 lists the error codes and the corresponding Boot Load function when running Boot Load from a disk cartridge.

Table 11-2 **Boot Load Error Codes—Disk Cartridge**

Code	Meaning
91	Cannot create the log file.
92	MIVolume has failed.
96	Failed to create a directory.
97	Check the log file. An error has occurred. The following are examples: - A ProductContent file is missing. - Cannot create a specified file. - Cannot append BTOS commands to BootLoad.sub. - A copy has failed.
98	Cannot access SysContent.
99	Fatal error—keyswitch in MANUAL position.

Note: *If, after initiating the System Load utility, the status code 30 is displayed but you have a valid release cartridge in the cartridge drive, you may have a bad drive. Contact your local Burroughs field support office.*

If the system displays a 20 and then immediately goes to a status of 99, the keyswitch may be faulty. Retry the procedure. If you get the same failure, contact your local Burroughs field support office.

Only the most recent error condition is indicated in the STATUS display. To get a complete listing of all errors that occurred during an unsuccessful Boot Load session, you must examine the Boot Load log file, [sys]<sys>BootLoad.log.

Normal Operations—Tape

The Boot Load utility uses the front panel STATUS display to indicate the progress of software installation. Table 11-3 lists the codes displayed and the corresponding Boot Load function for running Boot Load from a tape.

Table 11-3 **Boot Load Codes—Normal Operation with Tape**

Code	Meaning
80	Start Boot Load utility, create BootLoad.log file, and execute MChange Volume Name.
81	Pause for MIVolume. User must turn the keyswitch to NORMAL to initialize the system disk.
82	Initializing the system disk.
87	Restoring the system software from archive files on the release tape to the system disk.
90	Successful completion of the software installation from the currently loaded release tape.

Interpreting Error Conditions—Tape

If the Boot Load utility fails when running it from a tape, you will not be able to check the Boot Load log file to diagnose the problem. Also, if the Boot Load fails during BTOS system disk initialization or the restore operation, the STATUS display will hang at 82 or 87, respectively.

The BTOS system disk initialization should take no more than 45 minutes to complete. If you are initializing the BTOS system disk and the STATUS display shows 82 for more than 45 minutes, you probably have a faulty release tape, tape drive, or BTOS system disk.

Release files are stored on the Boot Load release tape in an archive file format. Therefore, instead of running a Copy utility, as is done with a disk cartridge, Boot Load executes a Restore utility on the release tape. The restore operation from the tape should not take longer than 20 minutes. If the STATUS display shows 87 for more than 20 minutes, you probably have a faulty release tape or tape drive.

Note: *Any 90 series code other than 90, 92, or 99, indicates an internal error. Contact your Burroughs support center for assistance.*

Checking the Boot Load Utility Log File

The Boot Load utility log file is [sys]<sys>BootLoad.log. The operations performed during the Boot Load session cannot be observed while the utility is executing. Therefore, you should examine the contents of this file using the ofed or ofvi editor to make sure that all of the operations were successfully executed. This can be done once you have loaded your workstation software.

When running the Boot Load utility from a release disk cartridge, the log file contains the following information:

- The volume name assigned to the BTOS system disk.
- The volume password assigned to the BTOS system disk.
- The parameters used to initialize the BTOS system disk, if this was done as part of the installation process.
- All of the BTOS operations executed during the Boot Load session as defined in the Boot Load Product Content file.
- Reports of any errors that occurred during the Boot Load session. Standard BTOS software error codes are used.

When running the Boot Load utility from a release tape, the log file contains a report of the MRestore utility that restored files from the release tape to the system disk.

Note: *If the BTOS system disk is being initialized as part of the Boot Load from tape, no Boot Load log file is created.*

Subsequent Boot Load sessions append their status messages to the log file.

When installing restricted mode software from a removable disk cartridge, a log file, [!d0]<sys>BootLoad.log, will also appear on the disk cartridge if the BTOS system disk was initialized as part of the software installation. The Boot Load log on the cartridge will contain the status of the MIVolume operation as well as that of the Boot Load utility.

Note: *If a previous session of Boot Load was terminated abnormally, it is possible that when you retry to run the utility, the system will display a 90 without displaying any other codes first. This means that no files have been installed.*

If this happens (that is, the system displays a 90 without displaying any 80 codes), turn the keyswitch to STOP and then back to REMOTE or NORMAL. The Boot Load procedure should run this time.

Interrupting Disk Initialization during Boot Load

If, during the Boot Load utility, you turn the keyswitch to STOP while an 82 is displayed, the badblk.sys file is corrupted. This file contains bad spot information about the BTOS system disk. If this file becomes corrupted, you could potentially have severe problems with your system disk. To recover, use the following procedure:

Note: *You can perform this procedure only if you know where your bad spots are on your system disk. A list of bad spots for each disk drive unit was originally supplied with your system. If you have previously installed software, you should have an updated list of the bad spots using the BTOS MVolume Report utility. If you do not have a list of the bad spots on your system disk, contact your local Burroughs Support Personnel for assistance.*

1 Turn the keyswitch to STOP.

2 Insert your Boot Load release medium into the drive. Restart the Boot Load utility, selecting initialization.

3 After the Boot Load utility is successfully completed, use the ofed or ofvi editor to edit the [sys]<sys>BootLoadParams file. Make the changes described in steps 4 through 8. (The format of the [sys]<sys>BootLoadParams file is described in Section 10.)

Caution: *Do not run the MSysLoad utility at this time.*

---

4 Edit the "Suppress format?" line so that it reads:

```
SuppressFormat,no
```

5 Edit the "Number of surface tests?" line so that it reads:

```
SurfaceTest,3
```

6 List your device type (Atasi46, Memorex166, and so on) on the "Device Type" line. If you do not know your device type, call your local Burroughs Support Personnel.

7 List the bad spots from the list that was provided with your system, or that were recorded when you ran the MVolume Report utility during a previous installation, on the "Bad Spots" line.

8 Exit from the editor.

9 Turn the keyswitch to STOP again.

10 Reinsert your Boot Load release medium into the drive. Restart the Boot Load utility, selecting initialization.

11 Your system disk is now back to normal. You can run the MSysLoad utility.

Note: *If you did not include the device type in the BootLoadParams file and attempt to run Boot Load, you will get a status code 92, MIVolume has failed. Perform the above procedure again, beginning at step 3.*

Diagnosing MSysLoad Utility Problems

Error Menus

If the MSysLoad utility encounters an error during the installation process, the following menu and prompt are displayed:

```
..................................................
..                                              ..
..    An error has occurred during installation.  ..
..                                              ..
..                                              ..
..    0 = Exit the MSysLoad Utility             ..
..                                              ..
..                                              ..
..    1 = Continue, with Error                  ..
..                                              ..
..                                              ..
..................................................

Please Select Whether to Continue or Exit ==>
```

To continue loading software despite the error, enter "1" and press the RETURN key. Depending on the type of error, the utility may be able to continue installing software from the medium. If so, you should check the MSysLoad log file to see the nature of the error.

To exit the MSysLoad utility, enter "0" and press the RETURN key. The following exit message is displayed:

```
System Load not Successful.  Please Check Log File.
```

Note: *If the utility encounters an error on a tape, the error menu is not displayed. The MSysLoad utility is automatically terminated, and this message is displayed:*

***ERROR. MSysLoad Cannot Recover from This Error. System Load not Successful. Please Check Log File.*

Refer to the following subsection for how to check the MSysLoad log file.

The MSysLoad Log File

When you exit from MSysLoad, the entire MSysLoad session that you just completed is appended to the BTOS log file [sys]<sys>SysLoad.log. This provides a record of the software that was installed during the session. Also, any installation error messages that occurred are contained in this file.

The session is denoted in the file by a banner that indicates the MSysLoad utility, and the date and time that the session took place.

Subsequent MSysLoad sessions append their status messages to this file.

If you are using the MSysLoad utility often, you may want to occasionally print out a copy of the log and then remove the log file. This practice saves memory space on your BTOS system disk.

The MSysLoad utility copies the messages displayed on the screen during the MSysLoad session to the log file. You can check this file to see what operations were executed during the installation procedure and to see if any error messages were displayed. The error messages are standard BTOS software status messages. Refer to the *BTOS Status Codes Reference Manual* and your *XE 500 CENTIX Administration Guide* for an explanation of BTOS status codes.

The Root/Swap Utility Log

A log of the user interaction during a Root/Swap utility session is kept in a file called /rootswap.log in the restricted mode root. All user interaction, along with any type of errors that may have occurred during the session, are recorded in this file.

For an initial CENTIX installation, the disk number, the partition number, and the size of the root partition and swap files are recorded.

When moving and/or resizing the root partition or swap files, the previous and current locations and sizes of the files are recorded.

The log file also contains messages about the processes that were executed to implement the selected functions. For example, the log would include messages indicating the following:

- The completion of creating a new root.
- Making the file systems for a new root.
- Moving the root.
- Deleting an old root.
- Mounting the new root.
- Performing a mklost+found.
- Unmounting the new root.
- File system checks on the root.
- Updating of the CENTIX configuration file.

Diagnosing CentixLoad Utility Problems

Error Menus

If the CentixLoad utility encounters an error during the installation process, an error message is displayed. After each error message, either an error menu or the message "Press GO or Return to Acknowledge Error" is displayed.

The error message provides a brief description of the error and often includes the corresponding BTOS or CENTIX error code. For a complete explanation of these error messages, refer to the subsection "CentixLoad Utility Error Messages" later in this section.

Error menus vary in form, depending on the type and severity of the error. The following error menu shows the maximum number of options that could be displayed

```
                    ERROR MENU

..................................................
..                                              ..
..        0. Exit CentixLoad                    ..
..                                              ..
..        1. Retry                              ..
..                                              ..
..        2. Go on to Next Medium               ..
..                                              ..
..        3. Go on to Next Product              ..
..                                              ..
..        4. Go on to Next File in Product      ..
..                                              ..
..................................................
```

Following each error menu is the prompt

```
Enter Selection ==>
```

The "Exit CentixLoad" option is always included. It is used to exit the CentixLoad utility.

The "Retry" option is included if the error is not serious and it may be possible to successfully complete the failing operation by retrying it.

The "Go on to Next Medium" option may be included. If selected, you are returned to the Main Menu, from which you can choose the next release medium to install.

The "Go on to Next Product" option may be included. If selected, the utility does not install the current product but advances to the next product on the medium.

The "Go on to Next File" in Product option may be included. If selected, the utility does not install the current file but advances to the next file in the Product.

Caution: *If the "Go on to Next File in Product" option is selected, the given product may not be loaded in its entirety and therefore may not function correctly.*

CentixLoad Utility Error Messages

The following error messages may be generated by CentixLoad when an installation error is detected. All error messages are both displayed at the terminal and entered in the CentixLoad log file.

Error messages normally have the format

I/O ERROR Centix errno= *n*

CENTIX message

CentixLoad message

or

I/O ERROR BTOS erc= *n*

BTOS message

CentixLoad message

The CENTIX and BTOS portion of the error message are generated by the system software when an I/O error has occurred. The CentixLoad portion of the error message is provided by CentixLoad as additional information.

If an error local to the CentixLoad utility occurs, only a CentixLoad message is generated.

The CentixLoad messages are listed below in alphabetic order. Each of the messages in the list is followed by a brief description of the error and possible corrective actions. For more information about CENTIX error codes, refer to your *XE 500 CENTIX Operations Reference Manual*. For more information about BTOS error codes, refer to the *BTOS Status Codes Reference Manual* and your *XE 500 CENTIX Administration Guide*.

Cannot append to [sys]<sys>Sys.Version

CentixLoad was unable to append to this BTOS file. The product at this point has been loaded; however, the entry in [sys]<sys>Sys.Version cannot be made. Verify that you have free space on your system disk (by using the MVolume Status utility through ofcli). Also try running the MFiles utility through ofcli as follows:

```
# ofcli -c "run <admin>MFiles.run,
[sys]<sys>Sys.Version,y"
```

If the MFiles utility successfully returns the size of this file, try running CentixLoad again.

Cannot determine size of [sys]<sys>Sys.Version

CentixLoad was unable to determine the size of this BTOS file. It will not be able to append to this file. The product at this point has been loaded; however, the entry in [sys]<sys>Sys.Version cannot be made. Try running the MFiles utility through ofcli as follows:

```
# ofcli -c "run <admin>MFiles.run,
[sys]<sys>Sys.Version,y"
```

If the MFiles utility successfully returns the size of this file, try running CentixLoad again.

This problem can be caused by a corrupted BTOS system disk.

Can't 'cd /'

CentixLoad could not change its directory to the root directory (/). This indicates a potentially serious error in the root file system. If you can "cd /," and **ls** and **pwd** indicate that you are in the root, rerun CentixLoad. If you cannot correctly "cd /," run **fsck** on /dev/root. If this does not correct the problem, try rebooting the system. If the problem still is not corrected, you may need to reinstall or restore the root partition.

Can't close CentixLoad.log

CentixLoad was unable to close the log file and flush its buffer. The log may be incomplete. This condition may indicate that the restricted mode's root file system is corrupted or filled.

Can't change directory to partition

CentixLoad has mounted a cartridge product from /dev/dsk/c0d0s7 as /.install, but was unable to "cd /.install." This could indicate problems in the root file system, but most likely it indicates that the disk cartridge partition is damaged.

Can't create: install

Can't write to: install

Can't remove: install

CentixLoad was unable to create the file /install. There may be problems in your restricted mode's root file system, the root file system may be filled, or there may be a file with that name that cannot be overwritten.

Can't create lock file

CentixLoad could not create the file /tmp/CentixLoadLock. This can be caused by a filled root file system, a missing /tmp directory, errors in the root file system, or other serious problems.

To verify that there is a problem, try creating a text file /tmp/CentixLoadLock. If you cannot create the file, the error may have been caused by one of the previously described problems. If you can create this file, remove the /tmp/CentixLoadLock file and rerun CentixLoad.

Can't create pipe

CentixLoad was unable to create a pipe for logging of the output from the install script. As a result the install script was not run. Try a CENTIX command such as

```
$ cat /etc/passwd | more
```

If this command runs successfully, try running CentixLoad again. If it does not, you may have too many processes or pipes currently open on the system.

Can't create process

CentixLoad was unable to fork a process to run the install script. As a result, the install script was not run. Try a CENTIX command such as

```
$ cat /etc/passwd | more
```

If this command runs successfully, try running CentixLoad again. If it does not, you may have too many processes running or may be running out of swap space.

Can't create: [sys]<sys>Sys.Version

CentixLoad determined that this BTOS file did not exist but it was unable to create it. The product at this point has been loaded; however, the entry to [sys]<sys>Sys.Version cannot be made.

Verify that you can create this file by copying a text file to it with the **ofcopy** command. If this works, remove the file and run CentixLoad again. If you are unable to create this file, the BTOS system disk may be out of space.

Can't get exit status of install script

CentixLoad was unable to successfully obtain the status of the install script via the **wait** system call. CentixLoad tries to kill the install script but does not report an error if it cannot.

Can't make directory for mount point

CentixLoad was unable to "mkdir /.install." This can be caused by a filled root file system, errors in the root file system, or other serious problems.

To verify that there is a problem, try creating the directory /.install with the **mkdir** command. If you cannot create this directory, the error has been caused by one of the previously described problems. If you can create this directory, remove it and rerun CentixLoad.

Can't open BTOS file: *filename*

This message is often preceded and/or followed by other messages which describe the cause or severity of the error.

Can't open CentixLoad.log - no log will be kept

CentixLoad is unable to open the file /CentixLoad.log. This condition may indicate that the restricted mode's root file system is corrupted or filled.

Can't remove installation mount point

CentixLoad was unable to "rm -r /.install" after unmounting /dev/dsk/c0d0s7. This usually means that another user has entered this directory.

Can't write to CentixLoad.log - log may be incomplete

CentixLoad was unable to write to the log file. CentixLoad only returns this error message once but continues to try to write to the log. This condition may indicate that the restricted mode's root file system is corrupted or filled.

CentixLoad terminated

CentixLoad was terminated by another user using the **kill** command.

[d0]<sys>Partition.7 renamed to restore missing BTOS file

A missing file was on a disk cartridge was recovered by CentixLoad. See "How the CentixLoad Utility Works" in Section 7 for more information.

This operation could occur if a fatal error occurred during a previous installation of software from the disk cartridge.

Device not present on the system

The device that was selected for loading software does not appear in the /dev file or is not on the system. Select another device and make sure that the release medium is in the device which you select.

Error accessing tape device

An I/O error occurred when trying to read from a release tape.

Error examining install script

Cannot determine what to append to [sys]<sys>Sys.Version

CentixLoad was unable to read the install script to determine what should be appended to [sys]<sys>Sys.Version. Try running CentixLoad again.

Error reported during installation

This indicates that the install script for the CENTIX product currently being installed returned an error exit status. The script should have reported the error just before exiting. If in doubt, you can select Retry from the error menu to reload the script and rerun it.

Extra Partition.7 on cartridge

There is more than one Partition.7 file on the disk cartridge. Check the Partition.7 files to see which is the valid one. Remove or rename the invalid files.

FATAL ERROR - CentixLoad will terminate

This message is preceded by an error message indicating what the error was. The only fatal errors are:

- Errors creating or accessing the CentixLoad lockfile.
- Errors setting up signal traps.
- Cannot change directory to the root directory (/).

fsck indicates errors in the partition

CentixLoad runs a file system check on each disk cartridge-based partition prior to loading it. The **fsck** command is run with the **-pq** options so that minor problems in the file system are fixed.

If this error occurs, the disk cartridge has probably been damaged. Try running CentixLoad again. If it fails again, replace the cartridge.

install script failed

The install script reported a nonzero exit status. The install script should have given a message explaining why it failed. If in doubt, try running the install script again.

install script killed

Due to an error while reading from the pipe, CentixLoad killed the install script so that it would not hang once the pipe was filled.

I/O error checking existence of BTOS file: *filename*

An I/O error occurred while trying to open and close a BTOS file to verify it's existence.

I/O error reading BTOS file: *filename*

This message is preceded and/or followed by messages describing the I/O error and its severity.

I/O error on mount

CentixLoad unable to mount a disk cartridge-based product from /dev/dsk/c0d0s7 as /.install. This usually indicates an error in the partition. The problem could also be caused by errors in the root file system.

Line too long in BTOS file: *filename*

This indicates that the given file has been damaged. You can attempt to correct the problem by editing the file, but you will probably have to replace the release medium.

Medium is in an unknown format

For a disk cartridge, the SysContent file is missing.

For a tape, the system cannot correctly read start-of-tape, the tape is blank, or the system is unable to cpio the install script into the CENTIX file system.

In all cases, the given medium cannot be loaded by CentixLoad. The release may be loadable by MSysLoad, but it is doubtful. Otherwise, the medium has been damaged, or the tape drive or disk cartridge drive is malfunctioning.

Medium is only loadable by MSysLoad

This indicates that the currently loaded release medium is not in the packaging format for CENTIX products, or that it is a BTOS product. You must run the MSysLoad utility to install the software products on that medium.

Medium is not mounted and on line

For a half-inch tape, make sure that the tape drive indicates that it is on line and loaded. Also verify that the tape drive is correctly connected to the SP or DP board. Also verify that the tape is in the correct tape drive.

For a disk cartridge, make sure that the cartridge is inserted and up to speed.

For a QIC tape, make sure that the QIC tape is inserted and on line.

Once you have corrected the previously described problems, you can select Retry from the error menu and continue to load software. If the medium still fails to come on line, make sure the device is properly configured into the system.

Missing BTOS file on cartridge: *filename*

The given file appears in the release disk cartridge's product content file or SysContent file, but is not on the cartridge. This error is not fatal unless the SysContent file is missing.

The product with a missing file cannot be completely loaded but other products on the cartridge can be loaded. In some cases the missing file may be recovered by CentixLoad by renaming Partition.7 to the missing file. See "How the CentixLoad Utility Works" in Section 7 for more information.

Must be a superuser

You must be a superuser in order to run CentixLoad.

Partition.7 is present, but can't determine what to rename it to

There is more than one missing file on the release disk cartridge and Partition.7 is present. CentixLoad is not able to load this cartridge. See "How the CentixLoad Utility Works" in Section 7 for more information.

If you want to load the products on the cartridge that do not have missing files, use the BTOS Rename utility to rename Partition.7 to something else and then rerun CentixLoad. To load the corrupted product(s) you must replace the disk cartridge.

Someone else is using CentixLoad

CentixLoad can only be run by one user at a time. If no one else is using CentixLoad, the lockfile (/tmp/CentixLoadLock) may not have been removed since the previous session.

CentixLoad tells you if it could not remove its lockfile, unless the session was terminated with a signal of 9 (sure kill). If you are certain that CentixLoad is not being run by someone else, use the **rm** command to remove this file yourself. Once you have removed this file, CentixLoad can be run again. Until you remove this file, CentixLoad continues to report that it is in use and does not allow you to load software.

SysContent file is damaged

The SysContent file on the release medium had a premature end-of-file or an I/O error. The medium is probably damaged and should be replaced.

SysContent or a Product Content file is damaged.

It specified BTOS file: *filename*

The SysContent file or a product content file on the release medium specified an invalid file name.

Tape not successfully loaded and installed

The software products on the release tape were not successfully installed. This message is preceded by errors messages that describe why the products could not be installed.

tee to log failed

An error occurred while CentixLoad was trying to read from the pipe in order to log the results of the install script.

Unable to close a BTOS file

An I/O error occurred while a BTOS file on the release disk cartridge was being closed. No recovery operations can be done. CentixLoad continues as if the error had not occurred.

Unable to initialize signal handling

The CENTIX system call **signal** reported an error. CentixLoad cannot run.

Unable to Load: *filename*

The named file could not be loaded. Other files in the product can still be loaded but the product may not be complete and so may not function properly. This message is preceded by error messages that describe why the file could not be loaded.

Unable to remove lock file: /tmp/CentixLoadLock

This indicates that CentixLoad was unable to remove the lockfile /tmp/CentixLoadLock. From the CENTIX shell, use the **rm** command to remove this file yourself. Once you have removed this file, CentixLoad can be run again. Until you remove this file, CentixLoad continues to report that it is in use and does not allow you to load software. Since this file is in the /tmp directory, it should be removed automatically when the system is rebooted.

Unable to rename BTOS file from, to:

oldfilename, newfilename

On a release disk cartridge, CentixLoad was unable to rename a product file to (or from) [d0]<sys>Partition.7. See "How the CentixLoad Utility Works" in Section 7 for more information.

This error affects the ability of CentixLoad to continue installing from the release disk cartridge. If this error occurs, it may be necessary to rerun CentixLoad and let it attempt the rename operation again.

Unable to unmount /dev/dsk/c0d0s7

This message indicates that the partition on a release disk cartridge could not be unmounted. Make sure that no users have changed their directory to the /.install directory on the cartridge or have unmounted the partition.

If the unmount still fails, you must exit CentixLoad. From the CENTIX shell, try unmounting the partition again. If it still fails, execute a "shutdown" and reboot the system in the mode in which it was previously running. Try loading software from the cartridge again. CentixLoad should report a missing file on the disk cartridge and report that is was restored.

Unable to unmount partition

This message indicates that the partition on a release disk cartridge could not be unmounted. Make sure that no users have changed their directory to the cartridge (/.install) or have unmounted the partition.

If the unmount still fails, you must exit CentixLoad. From the CENTIX shell, try unmounting the partition again. If it still fails, execute "shutdown," remove the disk cartridge, and reboot the system. Try loading software from the disk cartridge again. CentixLoad should report a missing file on the disk cartridge and that it restored the missing file.

The CentixLoad Utility Log

A log of the user interaction during a CentixLoad session is kept in a file called /CentixLoad.log in your root directory. All user interaction, along with any type of errors that may have occurred during the session, are recorded in this file. Subsequent CentixLoad sessions append their status messages to this file.

The log file contains the date and time that the CentixLoad session began and the operations it performed. This includes

- The type of medium from which a product was loaded.
- Any error messages that were returned by CentixLoad.
- The user's selection of error recovery.
- Which files or products were loaded or skipped.
- Messages logged and files copied by a product's install script.

Recovering from Improperly Initialized System Disk

Prior to initializing the system disk, the Boot Load utility checks the system disk device to see if it has already been initialized. If it has, the Boot Load utility can use information in the disk's volume control structure files to determine what type of disk device it is. The utility can then reinitialize the disk with the proper parameters.

If the disk has not been initialized before being installed into the XE 500, the Boot Load utility cannot tell what type of device it is. The Boot Load utility initializes the disk with a default set of device parameters. These parameters cause the disk to be initialized for only about 17 MB, which is lower than the amount of memory space actually available on the disk. The system disk must be reinitialized through the Boot Load utility and the workstation software must be installed again.

Perform the following procedure to reinitialize the system disk properly.

1 Use the ofed editor to edit the BTOS BootLoadParams file, [sys]<sys>BootLoadParams. (For information on how to use the ofed editor, refer to volume 2 of the *XE 500 CENTIX Operations Guide*.)

2 After the entry "Device" in the file, enter a comma and then the proper device type name. Valid names are

- ATASI46 for a 37.5 MB built-in disk.
- MICROPOLIS85 for a 71.3 MB built-in disk.
- TOSHIBA85 for a 72.2 MB built-in disk.
- MEMOREX166 for an SMD disk.

3 After the entry "BadSpots" in the file, enter a comma and the list of bad spots. Use only one space between bad spot entries; do not use carriage returns.

Caution: *Any syntax error in the BootLoadParams file will cause the Boot Load utility to hang with an 82 in the STATUS display when it is initializing the system disk. If this occurs, the BootLoadParams file must be corrected and the Boot Load utility run again.*

---

4 Exit the editor.

5 At the CENTIX superuser prompt, enter

```
# exit
```

and press the RETURN key. You are returned to the Restricted Mode Main Menu.

6 At the Restricted Mode Main Menu prompt, enter "0" and press the RETURN key. This shuts down the CENTIX system.

7 When the message saying that you can reset the processor is displayed, put the Boot Load release medium on-line.

8 Turn the front panel keyswitch to STOP and then to REMOTE. This invokes the Boot Load utility and reinitializes the system disk with the proper formatting parameters.

9 Because all data on the system disk is overwritten when it is reinitialized, you must repeat the rest of the Boot Load utility procedure and reinstall the BTOS software.

Overview of System Configuration Files

XE 500 BTOS uses several text files to define the system's configuration, both hardware and software. These text files are read when BTOS is loaded into the XE 500 BTOS-based processor boards (that is, when the system is *booted*).

Note: *Different sets of configuration files are read for the different positions to which the keyswitch can be turned to boot the system. See your* XE 500 CENTIX Administration Guide for details.

There are several types of configuration files:

1 The *master configuration file* is used by the master processor (FP00 or DP00) to determine the processor operating systems to be loaded at boot time.

  The master configuration file has an entry for each processor in the system, except the master processor. The entries define the version of BTOS or CENTIX to be loaded on the processors. Other entries in this file define the processor configuration files for applicable processors.

2 A *processor configuration file* defines the hardware and software configuration associated with a processor board.

  For example, a File Processor (FP) or Disk Processor (DP) configuration file lists the device names and passwords for each disk drive under its control. A Cluster Processor (CP) or Terminal Processor (TP) configuration file defines parameters for a CP's or TP's input/output (I/O) ports.

  Each configuration file also has entries that define the dynamic block allocation parameters for the processor. These parameters are used to determine the size of I/O communications buffer space.

  Applications Processors (APs), which run CENTIX, do not have configuration files.

3 A *processor initialization file* defines the system services that are to be run on a processor once that processor has booted up. Entries in this file are normally the run files of the desired system services.

  AP processors do not have initialization files.

4 The *CENTIX configuration file*, [sys]<sys>ConfigUFS.sys, lists the BTOS partition files that correspond to logical CENTIX disk devices. A CENTIX disk device, which is actually the disk extent assigned to a BTOS partition file, is usually setup as a CENTIX file system.

5 The *queue index file* defines the BTOS queues for which the BTOS Queue Manager is responsible.

6 The *lpr spooler configuration file*, [sys]<sys>splcnfg.sys, contains entries for each printer in the system that can support the CENTIX lpr spooler. A printer entry defines parameters such as where the printer is connected and what print queue the printer is to serve. A default version of this file comes with the standard software.

   The lpr spooler is a BTOS-based printer spooler.

7 The *lp spooler configuration file*, [sys]<sys>LpSpooler.config, defines the lp spooler services for BTOS. This file comes with the standard software and should not be modified.

   The lp spooler is a CENTIX-based printer spooler. Refer to the *XE 500 CENTIX Administration Guide* for details on configuring and running the lp spooler.

8 *I/O device configuration files* define the hardware and software parameters of disk drives, printers, tape drives, and modems.

   The necessary disk drive configuration files are included in the standard software release. Default configuration files for a parallel printer, a serial printer, a half-inch tape drive, and a QIC tape drive are also provided. The printer and tape drive configuration files can be modified and other I/O device configuration files can be created by using the BTOS utility MCreate Configuration File.

This section describes the configuration files related to the XE 500 processor boards, the CENTIX configuration file, and I/O device configuration files. The processor initialization files are described in Section 13. The queue index file is described in Section 19. Section 20 describes how to create I/O device configuration files.

The Master Configuration File

At boot time, the master processor (FP00 or DP00) loads its own operating system, [sys]<sys>SysImage.sys. (The operating system is also referred to as a *system image*.) It then reads the default master configuration file, [sys]<sys>Master.cnf, to determine the versions of BTOS or CENTIX for the other processor boards and the order in which to load them.

When booting the system in restricted mode (for example, during Boot Load), the master configuration file [sys]<sys>Master.r.cnf is used. This special master configuration file allows enough of the system to be brought up so that the MSysLoad and MBTOS Config utilities can be run.

The master configuration file comprises three types of line entries:

1 The nowatchdog entry.

2 Processor BTOS entries.

3 Include statement entries for processor configuration files.

The Nowatchdog Entry

The line

```
nowatchdog
```

must be included in the master configuration file.

Processor Operating System Entries

Following the nowatchdog entry are operating system entries for each processor in the XE 500 system, except the master processor. Each of these entries determines the version of BTOS or CENTIX to be run on the processor. Entries are ordered according to how their corresponding processors are configured in the XE 500 enclosures. The first entry corresponds to the first processor board after the master processor, the next entry to the next processor board, and so on.

Include Statement Entries

The master configuration file also contains an *include* entry for the processor's configuration files.

A CP or TP configuration file defines the I/O ports supported by the processor and the X, Y, and Z block allocations. An FP or DP configuration file defines the Y and Z block allocations and the internal disk devices associated with the processor. An SP configuration file defines the Y and Z block allocations for the SP. The AP has no configuration file.

Caution: *The master processor and SP configuration files should not be included in the master configuration file. The system automatically assumes the master processor configuration file to be [sys]<sys>Fp00.cnf or [sys]<sys>Dp00.cnf, and all SP configuration files to be [sys]<sys>Sp*nn*.cnf, where* nn *is the SP number.*

By convention, processor configuration file names always have the suffix ".cnf."

Sample Master Configuration File

A sample master configuration file is shown below:

```
NoWatchDog
Cp              [sys]<sys>CpBtos.sys
Include         [sys]<sys>Cp00.cnf
AP              [sys]<sys>Centix.sys:[sys]<sys>Centix20.sys
Tp              [sys]<sys>TpBtos.sys
Include         [sys]<sys>Tp00.cnf
Dp              [sys]<sys>DpBtos.sys
Include         [sys]<sys>Dp00.cnf
Sp              [sys]<sys>SpBtos.sys
Fp              [sys]<sys>sysimage.sys
Include         [sys]<sys>Fp01.cnf
Cp              [sys]<sys>CpBtos.sys
Include         [sys]<sys>Cp01.cnf
```

This master configuration file is for a system that contains, in ascending order by slot number, a master FP, a CP, an AP, a TP, a DP, an SP, a second FP, and a second CP.

The BTOS version entries comprise the processor name in the first column and the processor's BTOS version file in the second column. The master processor, and all processors of the same type as the master, run the master version of BTOS, [sys]<sys>SysImage.sys. BTOS versions for all other processors take the form [sys]<sys>*Xp*Btos.sys, where *Xp* represents the processor type (CP, TP, SP, FP, or DP).

Note that each BTOS version entry is followed by an include entry for that processor's configuration file (except for the SP, as discussed in previous cautionary note). The include entries are denoted by the word "Include" in the first column, followed by the processor's configuration file. Processor configuration files take the form [sys]<sys>*Xpnn*.cnf, where *Xp* represents the processor type and *nn* is the processor's number within the system. This number is used to uniquely identify processor boards of the same type.

The AP entry contains two operating system file names, one for both types of APs. The operating system [sys]<sys>Centix.sys is installed on the API boards. The operating system [sys]<sys>Centix20.sys is installed on APII boards.

Note: *The master configuration file must have tab characters, not spaces, between the first and second column entries.*

Processor Configuration Files

A processor configuration file defines the protocols and/or hardware configuration of a processor's I/O ports. It also includes the X, Y, and Z block allocations for the processor.

In addition to defining X, Y, and Z block allocations, processor configuration files for FPs and DPs define the disk drives that they support and for CPs and TPs define the terminal ports that they support. By convention, processor configuration files always have the suffix ".cnf".

Processor configuration file names take the form [sys]<sys>*Xpnn*.cnf, where *Xp* is the processor type (For example, FP, CP) and *nn* is the processor number (for example, 00, 01, or 02). For example, the first CP is CP00, the second CP is CP01, the third CP is CP02, and so on.

See Appendix B for a listing of default processor configuration files.

Dynamic Block Allocation

X, Y, and Z blocks are used to provide buffer space for various types of communications.

X blocks are used exclusively in the CP for data transmission to and from PT 1500s. The number of X blocks affects both the ability of PT 1500s to boot and PT 1500 performance after booting. The default size of an X block is 2560 bytes.

Y blocks are the large IPC (Inter-Processor Communications) communication blocks. They generally handle responses from a server process to a client process, should a large amount of data have been requested. The default size of an Y block is 2560 bytes.

Z blocks are the small IPC communications buffers, and are used for the same purpose as Y blocks, but are smaller. The default size of a Z block is 100 bytes.

Note: *The default values provided for the dynamic block allocations allow adequate performance for most configurations.*

X, Y, and Z blocks are allocated by the addition of a line (or lines) in the processor configuration file. These line entries have the following form:

Blocktype=*type*, Number=*nn* [, Size=*ss*]

where

Blocktype	must begin in column 1.
type	is "X" to allocate X blocks, "Y" to allocate Y blocks, and "Z" to allocate Z blocks.
nn	is the number of blocks to allocate.
ss	is the size of the blocks to be allocated. This parameter is optional. If it is not specified, the default size is used.

For Y and Z blocks, if the number of blocks times the size of the blocks to be allocated is larger than the space configured for a particular block type when the processor's operating system was generated, new space is dynamically allocated. This new space is used instead of the space allocated by the operating system. Otherwise the space configured when the processor's operating system was generated is used. (Default operating systems are generated when the software release is being packaged. The parameters used by a BTOS operating system can be modified by customizing the operating system. Refer to your *XE 500 BTOS Customizer Guide*.)

For X blocks, if there are values in the configuration file for the number of X blocks or the size of the X blocks, then those values are used to allocate space. Otherwise, the space configured when the CP's operating system was generated is used.

See Appendix B for a listing of default processor configuration files.

Note: *Default X, Y, and Z block entries are automatically included when you create default processor configuration files using MBtos Config. Refer to Section 14.*

File and Disk Processor Configuration Files

An FP or DP configuration file defines the processor's dynamic block allocation and lists the internal disk devices that are connected to it. The FP or DP reads this file after its operating system has been loaded and it has booted up.

File Entries

There are two entries in this configuration file that define the number of Y and Z blocks to be allocated.

There is one entry for each disk drive connected to the processor. These entries define the device names and passwords assigned to the disk drives.

The following lines represent the entries of a default DP00 configuration file:

```
BlockType=y, Number=18
BlockType=z, Number=40
DISK 0, DEVICE=s0, PASSWORD=s0
DISK 1, DEVICE=s1, PASSWORD=s1
DISK 2, DEVICE=s2, PASSWORD=s2
DISK 3, DEVICE=s3, PASSWORD=s3
DISK 4, DEVICE=s4, PASSWORD=s4
DISK 5, DEVICE=s5, PASSWORD=s5
```

The first two lines define the number of Y and Z blocks allocated (18 and 40, respectively).

The next six define disk drive parameters. This DP supports six SMD disk drives. The first column of a disk drive entry lists the disk drive location. The next two columns define the drive's device name and password. In this sample, the device names and device passwords for SMD disk drives 0, 1, and 2 are s0, s1, and s2, respectively.

The following lines represent the entries of a default FP00 configuration file:

```
BlockType=y, Number=18
BlockType=z, Number=40
DISK 0, DEVICE=d0, PASSWORD=d0
DISK 1, DEVICE=d1, PASSWORD=d1
DISK 2, DEVICE=d2, PASSWORD=d2
DISK 3, DEVICE=d3, PASSWORD=d3
```

The first two lines define the number of Y and Z blocks allocated (18 and 40, respectively).

The next four define disk drive parameters. This FP supports one cartridge and three built-in disk drives. The first column of a disk drive entry lists the drive's location. The next two columns define the drive's device name and password.

Disk Device Configuration Used by the System

The system's disk configuration definition is determined by two factors: (1) the parameters used when the FP or DP operating system was generated and (2) the contents of the processor configuration file.

When an FP or DP operating system is generated, Device Control Blocks (DCBs) are created for the number of disks defined by "nWinch," which is one of the parameter entries used during system generation. (For detailed information about operating system parameters for BTOS, refer to the *XE 500 BTOS Customizer Operations Guide*.) This entry defines the disk number, device name, and device password for each disk controlled by the FP or DP.

Disk information as defined in the processor's operating system can be modified by the information for that disk's entry in the FP's or DP's configuration file. However, if the disk's entry in the configuration file is deleted, the information in the processor's operating system is used.

The disk parameters in the standard FP and DP operating systems match those defined in the standard FP00 and DP00 configuration files. However, to insure unique device names, the configuration files for all FPs and DPs other than FP00 and DP00 include entries that redefine the device information. For this reason, you should not delete configuration file disk entries for FP01 and up, and DP01 and up.

For example, while the standard operating system for FP01 defines its disk device names to be d0 through d3, the FP01 configuration file redefines them to be d4 through d7. If, say, the entry for d4 was deleted from this file, the default information for d0 would be used. This would cause a duplicate device name, and erc 236 (Invalid device specification) would be returned when the system attempted to mount disk d4.

Cluster and Terminal Processor Configuration Files

The CP and TP configuration files (for example, [sys]<sys>Cp00.cnf for CP00) are used to assign the operating parameters for the CP's and TP's RS-232-C ports and the baud rates for the CP's two RS-422 ports. Serial printers, terminals, remote PT 1500s, and modems connect to RS-232-C ports. PT 1500 cluster lines connect to CP RS-422 ports.

A CP or TP configuration file must contain an entry for each port to be used.

The CP and TP configuration files are also used to define the dynamic block allocation for the processors.

The last entry of a CP configuration file defines the number of PT 1500s that are connected to the CP.

Caution: *A CP configuration file must have at least one async entry for the CP to boot.*

RS-232-C Port Line Entries

Entries for the RS-232-C ports take the following form. Square brackets are used here to indicate optional parameters.

[async] *ChannelNumber*, [Parity=*n*], [Speed=*n*], [StopBits=*n*], [CharBits=*n*], [FlowGen], [FlowAct], [FlowAny], [XOFFlev=*n*], [XONLev=*n*], [modem], [connect=ctos]

where

async	determines that the port run in async mode. If "async" is not included in port entries for CP channels 1 through 2 and TP channels 1 through 4, these ports will run in sync mode. CP channel 3 and TP channels 5 through 10 are async only ports and so "async" is always included in their port entries.
ChannelNumber	is the processor channel number corresponding to the RS-232-C port, CP channels 1 through 3 and TP channels 1 through 10.
Parity=*n*	is the parity setting for data transmitted through the port. Valid values for *n* are "none," "even," or "odd." If this parameter is not included, the default parity is none.
Speed=*n*	is the baud rate for the port. Valid values for *n* are: 50, 75, 110, 134.5, 150, 200, 300, 600, 1200, 1800, 2400, 4800, or 9600. Sync/async channels can also run at 19200.

StopBits=*n*	is the number of stop bits for data transmitted through the port. StopBits=1 and StopBits=2 set the number of stop bits to 1 and 2, respectively. StopBits=0 sets the number of stop bits to 1.5.
CharBits=*n*	is the number of character bits per data character being sent through the port. Valid values for *n* are: 5, 6, 7, or 8. You cannot use values less than 7 in conjunction with the flow control parameters.
FlowGen	if included, specifies that the processor should issue XON and XOFF characters, as appropriate, during data transmission. If this parameter is not included, XON and XOFF characters are not issued.
FlowAct	if included, specifies that the processor should honor incoming XON and XOFF characters during data transmission. If this parameter is not included, XON and XOFF characters are not honored.
FlowAny	if included, specifies that, after receiving an XOFF, the processor should honor the next character received during data transmission as an XON. If this parameter is not included, XON and XOFF characters are honored according to the status of the FlowGen and FlowAct parameters.
XOFFlev=*n*	specifies that if the number of free slots in the character input FIFO (first in, first out buffer) falls below *n*, an XOFF should be sent. This parameter is only valid if the FlowGen parameter has been specified.
XONLev=*n*	specifies that if the number of free slots in the character input FIFO increases to *n* and an XOFF was previously sent, an XON should be sent. This parameter is only valid if the FlowGen parameter has been specified.
modem	specifies that the XE 500 not transmit on the line until the appropriate modem signals are detected.
[connect=ctos]	specifies that the port is to be used as the CLI port. An RS-232-C serial terminal can be connected to this port. The CLI can then be run interactively from this terminal. Only one RS-232-C port per CP or TP can be configured as the CLI port. For a CP, it must be channel 3; for a TP, it must be channel 10.

Note: *The line characteristics for an RS-232-C line supporting a printer are not defined by the entry in the processor configuration file. The characteristics defined in the printer's configuration file are used instead. There must, however, still be an entry for the line in the processor configuration file.*

RS-422 Cluster Port Line Entries

Entries for the CP RS-422 cluster ports take the following form. Square brackets are used here to indicate an optional parameter.

LINE *LineNumber* [, Clock=*n*]

where

LineNumber	is the number of the RS-422 port. Valid numbers are 1 (for cluster lines 1A and 1B) and 2 (for cluster lines 2A and 2B).
Clock=*n*	specifies the baud rate for the port. Valid values for *n* are 307k (for a baud rate of 307 k) or 2M (for a baud rate of 1.8 M). If this parameter is not included, the default value of 307k is used. PT 1500s can only run at 307 k baud.

PT 1500 Entry

The last entry of a CP configuration file defines the number of PT 1500s that are connected to the CP. This entry takes the form

pt *n*

where *n* is the number of PT 1500s. The maximum value for *n* is 16.

Default Cluster Processor Configuration File

The following lines represent the entries of a default CP configuration file:

```
BlockType=x, Number=24
BlockType=y, Number=5
BlockType=z, Number=10
async 3, speed=9600, parity=none, stopbits=1, charbits=8
line 1, clock=307k
line 2, clock=307k
pt 1
```

This CP is supporting one serial printer, connected to channel 3, and two PT 1500 cluster lines. The cluster lines are set to run at a baud rate of 307 k. Support for one PT 1500 is provided.

Storage Processor Configuration File

The SP configuration file contains dynamic block allocation entries for Y and Z blocks. Following is a default SP configuration file:

```
BlockType=y, Number=4
BlockType=z, Number=30
```

The number of Y and Z blocks allocated are 4 and 30, respectively. Note that each SP must have a configuration file for the SP to boot properly, even if the file does not contain any line entries.

CENTIX Configuration File

All I/O operations to XE 500 devices are made through software running on BTOS-based processors. The CENTIX operating system has its own programs and conventions for handling I/O operations to peripheral devices. However, because CENTIX I/O requests must eventually be handled by BTOS software, a connection must be made between the CENTIX *logical* devices and the actual BTOS-controlled devices. (Each CENTIX logical device is assigned a CENTIX minor device number.)

The CENTIX file system configuration file has entries that establish the connection between some CENTIX logical I/O devices and their corresponding BTOS-based devices. These devices include

- Disk devices.
- Tape drives.

Note: *The CENTIX configuration file uses tab characters, not spaces, to separate the columns of each entry.*

Disk Device Entries

Disk device entries establish the connection between a CENTIX logical disk device and a BTOS partition file.

Each CENTIX logical disk device is assigned a minor device number. This minor device number is internally linked to the logical disk devices listed in the CENTIX /dev directory (for example, /dev/dsk/c0d0s0, /dev/dsk/c0d0s1, and so on). A CENTIX logical disk device, which is actually the disk extent assigned to a BTOS partition file, normally contains a file system. The file system can then be made accessible to users by mounting it to a directory.

Refer to Appendix D for tables that explain the device naming conventions that are new with the 6.0 release and their relation with the previous conventions.

A BTOS partition file can exist for each CENTIX disk device. The disk extent allocated by BTOS to a partition file is then used to simulate a disk device for CENTIX. Because of this method, each actual XE 500 disk can "contain" multiple CENTIX logical disk devices.

The standard release version of this file supports a maximum of eight partition files per disk drive, excluding any swap partitions used by the APs in the system. It includes an entry for each possible BTOS partition file that could exist on XE 500 disks d0 through d3 and SMD disks s0 through s5.

Partition and Swap File Names

The file names for the partition files have the format

[*disk*]<sys>partition.*n*

where *disk* is the disk device name and *n* is the partition number, from 0 to 7.

Swap file names have the format

[*disk*]<sys>Swap.Ap.*nn*

where *disk* is the disk device name and *nn* is the number of the AP to which the swap area is assigned.

Disk Device Entry Format

Each partition file entry is divided in four columns.

1 The first column is used to indicate root partitions and swap files.

  The word "root" appears in this column if the partition is the CENTIX root partition. The word "swap" appears in this column if the file is a CENTIX swap file.

2 The BTOS partition or swap file name is listed in the second column.

3 The third column of an entry indicates the CENTIX File System Server that is assigned to handle disk I/O operations for that partition or swap file. The actual entry here is the name of the processor on which the CENTIX File System Server runs.

4 The fourth column lists the CENTIX minor device number for the logical CENTIX disk device.

Tape Device Entries

The CENTIX file system configuration file contains tape device entries for each tape device in the system. These entries define the correspondence between the CENTIX minor device number for a tape device and the BTOS name for that device.

The standard release version of the file includes the following tape entry for a QIC tape drive and a half-inch tape drive:

```
TAPE        [qic]                                   000
TAPE        [tape]                                  001
```

The first column entries indicate that the line entry is for a tape device.

The second column entries list the BTOS names for the respective tape devices ([qic] for the QIC tape drive, [tape] for the first half-inch tape drive connected to the first SP or DP in the system).

The third column is blank.

The fourth column entries list the CENTIX minor device numbers for the tape drives. These minor device numbers are internally linked to the logical tape devices listed in the CENTIX /dev directory (that is, /dev/rmt/c0d0 for the QIC tape drive, /dev/rmt/c1d0 for the first half-inch tape drive).

I/O Device Configuration Files

XE 500 disk drives, system printers, half-inch magnetic tape drives, QIC tape drive, and data communication (data comm) equipment (for example, modems) require configuration files to define their hardware and software operating parameters.

Disk Drive Configuration Files

The necessary disk drive configuration files are included in the standard software release. Disk drive configuration file names take the form

[sys]<sys>*drivename*.cnf

where *drivename* is the name that specifies the drive type for the system. These files do not normally require modification. The disk drive configuration files that come with standard software are listed in Appendix B.

lpr Printer Configuration Files

Default configuration files for a parallel printer and a serial printer to support the CENTIX lpr print queue are included in the standard software release. The parallel printer configuration file name is

[sys]<spl>SplConfigCp00.cnf

The serial printer configuration file name is

[sys]<spl>SplCConfigCp00.cnf

Printer configuration files are referenced for the system in the printer spooler configuration file, [sys]<spl>SplCnfg.sys. Because the printer spooler configuration file can be modified, printer configuration files can have any file name. Also, the same printer configuration file can be used for more than one printer.

For serial printers, the line characteristics defined in the printer's configuration file override those defined in that line's port entry in the processor configuration file.

Printer configuration files can be modified or created by using the MCreate Configuration File utility. Refer to Section 20 for instructions on how to use this utility.

Half-Inch Tape Drive Configuration Files

A default configuration file for a half-inch tape drive is included in the standard software release. The default half-inch tape drive configuration file name is

[sys]<sys>TapeConfig.sys

Half-inch tape drive configuration files can be modified or created by using the MCreate Configuration File utility. Refer to Section 20 for instructions on how to use this utility.

QIC Tape Drive Configuration File

The necessary QIC tape drive configuration file is included in the standard software release. The file name is

[sys]<sys>QicConfig.sys

This file does not normally require modification. If you must modify this file, use the tape option of the MCreate Configuration File utility. Refer to Section 20 for instructions on how to use this utility.

Communication Device Configuration Files

Configuration files for communication devices, such as modems or terminals, must be created using the MCreate Configuration File utility. Refer to Section 20 for instructions on how to use this utility. Refer to the device's supporting documentation to determine parameter values for which the MCreate Configuration File utility will prompt you.

BTOS System Services

XE 500 BTOS provides system services that control access to input/output (I/O) devices and BTOS-based software. These system services run on XE 500 BTOS-based processor boards. Default system services are installed on the appropriate processor boards as part of the system software configuration procedure.

System services include

- The AdminAgent.
- The Queue Manager.
- lpr small printer spooler manager.
- lpr big printer spooler manager.
- lp printer spooler server.
- Tape server (for half-inch tape drive).
- QIC server (for QIC tape drive).
- CENTIX file system server.
- Servers for application software products.

The BTOS software uses processor initialization files to install these services onto appropriate processor boards. These initialization files are read when BTOS is loaded into the XE 500 BTOS-based processor boards (that is, when the system is *booted*). The AP does not have an initialization file.

This section describes the processor initialization and system services configuration files. Section 14 describes how they are created and modified to suit your system's configuration.

A general description of processor initialization files is provided first. The system services are described in the subsequent subsections. Each system service description includes a discussion of the run statement that installs that service into a processor.

Processor Initialization Files

Each processor except the AP normally has an initialization file. Processor initialization file names take the form [sys]<sys>init*Xpnn*.jcl, where *Xpnn* is the processor designation (for example, FP00, CP02).

The initialization file is a text file that has a line entry for each system service to be run on the processor. Each line entry in an initialization file is a run statement that loads and activates the system service's run file on the processor.

Format for Run Statements

The processor initialization file is a subset of a type of file called a *JCL (Job Control Language) command file*. The Command Line Interpreter (CLI) is a program that reads and executes the statements in a JCL command file.

A run statement is one of the types of statements that can be included in a JCL command file. The format for a run statement in a JCL file is

```
$run runfile, parameters
```

where

runfile	is the full path name of the desired run file.
parameters	is a list of parameters related to the run file. Parameters are separated by commas.

Each processor initialization file comprises only run statements. One or more run statements are used to load and activate a given system service on the processor.

At boot time, the CLI is activated and reads its input from the processor initialization files. The master utilities whose run files are included in an initialization file execute on that processor, thereby installing the corresponding system services.

Creating Partitions

A BTOS processor's onboard memory can be divided into regions called *partitions* (these onboard partitions should not be confused with the BTOS partition files associated with the CENTIX file system). A program being run by the processor can be assigned a partition in which to execute. The program restricts its instructions and data to this assigned area of memory. This allows the processor to operate in a *multitasking* mode, in which several programs are running on the processor at the same time. Each program is actually sharing processing time on the processor.

There are two types of BTOS processor memory partitions:

- A *primary partition* is automatically set up for each processor at boot time. Any services that require user interaction are run in the processor's primary partition. Also, to ensure optimum system performance, some system services must be run in a primary partition.
- *Secondary partitions* can be created by including a run statement in a processor initialization file that installs the partition at boot time. Because secondary partitions do not directly support interactive input/output (I/O), programs that require user interaction must be run in a primary partition.

  Some system services that do not require user interaction can be installed in secondary partitions.

Some of the standard release system services are preconfigured to be installed in secondary partitions. If new system services are to be installed after your system is delivered, you may want to create secondary partitions for them to enhance overall system performance. Section 14 provides guidelines and procedures for installing system services.

Creating Secondary Partitions at Boot Time

Note: *When creating secondary partitions, the remaining memory in the primary partition must be at least the size of the new secondary partition plus have room for the CLI to be loaded after BTOS MCreate Partition utility has completed executing. This means that, to add a secondary partition to a processor, the processor must have 768 kB of memory.*

Some of the standard release BTOS system services are preconfigured to be installed in secondary partitions. If new system services are to be installed after your system is delivered, you may want to create secondary partitions for them to enhance overall system efficiency.

To have a secondary partition for a processor be created automatically at boot time, use the ofed or ofvi editor to include the following run statement in the processor's initialization file:

$run [sys]<admin>MCreatePartition.run,*size*,*name*

where

size is an optional parameter that determines the size of the partition when it is created. The parameter value is the size of the partition, in kilobytes (kB), followed by the letter "k". The size of a partition is limited only by available memory. If this parameter is not specified, the default value "200k" is used.

In addition to the space required by an application program, the partition needs 1.5 kB to store system data structures associated with it. MInstall Server requires that the minimum size of a secondary partition be 6 kB.

name is an optional parameter that determines the name of the secondary partition. If this parameter is not specified, the default name of the partition is background*nn*, where *nn* is 00 the first time the utility is invoked on the processor, 01 the second time, and so on. The partition name must not exceed 12 characters.

It is strongly recommended that partition names contain only alphanumeric characters. Only names with alphanumeric characters will be displayed by other programs.

The partition will be created the next time the system is booted.

You can prevent one of these secondary partitions from being created at boot time by deleting the appropriate run statement from the processor initialization file.

You can obtain status information about any secondary partition by executing the BTOS MPartition Status utility (refer to the *XE 500 BTOS Operations Reference Manual*).

Installing Applications at Boot Time

Applications can be installed into secondary partitions by including a run statement in a processor initialization file that installs the service or application into a secondary partition at boot time.

The MInstall Server run file installs a service or application into the secondary partition of a processor. The MInstall Server calls another utility, MInstall Secondary ([sys]<sys>MInstallSecondary.run) as part of the partition installation operation.

To have a service or application installed into a secondary partition at boot time, use the ofed or ofvi editor to add the MInstall Server run statement in the appropriate processor's initialization file. The MInstall Server run statement must follow the run statement that creates the secondary partition.

The MInstall Server run statement has the form

$run [sys]<admin>MInstallServer.run,*PartitionName*, *SvrRunFile*

where *PartitionName* is the name of the secondary partition and *SvrRunFile* is the run file, including any parameters, of the service or application to be installed. The application will be loaded into the partition the next time the system is booted.

For example, the following two run statements would cause the QIC server to be installed into the secondary partition called MQicServer with a size of 60 kB:

```
$run [sys]<admin>MCreatePartition.run,60k,MQicServer
$run [sys]<admin>MInstallServer.run,MQicServer,
     [sys]<admin>MQicServer.run
```

If the server fails to install at boot time, an error message is logged in the BTOS system log, [sys]<sys>log.sys, which can be examined with the BTOS MPLog utility (refer to the *XE 500 BTOS Operations Reference Manual*).

You can remove these applications by deleting the appropriate run statements from the processor initialization file.

AdminAgent

The AdminAgent is a system service that manages the execution of BTOS utilities on XE 500 processors. AdminAgent can run in one of two modes—single or multiple. For more information about AdminAgent, refer to Section 17.

Queue Manager

The Queue Manager controls the various BTOS-based queues in which related tasks are stored for processing. For example, the Queue Manager controls the queues that are used to handle lpr print requests. It can also manage similar queues for data communications processes and user-defined processes.

For more information about the Queue Manager, refer to Section 18.

lpr Printer Spooler Managers

The lpr printers connected to an XE 500 processor board (either a CP or TP) are controlled by an lpr printer spooler manager running on that processor. The lpr printing queue is controlled by the XE 500 Queue Manager.

There are two different lpr printer spooler managers available for the XE 500. One is called the "lpr printer spooler manager" or the "small lpr printer spooler manager," and the other is called the "big lpr printer spooler manager."

The small lpr printer spooler manager can support one parallel printer and up to three serial printers. It is designed to run on a CP or TP. If running on a TP, the small printer spooler can only control printers connected to serial Channels 1, 2, and 3.

Notes:

1 *A small lpr printer running on a 256-kB CP can support only two printers total. To support four printers, the CP must have 768 kB of processor memory.*

2 *If you want to connect serial printers to TP serial channels 4 through 10, the big lpr printer spooler must be installed on the TP.*

The big lpr printer spooler manager can support a parallel printer and up to 10 serial printers. It can only run on a TP.

Except for the number of printers supported, the features, capabilities, and installation of the managers for the two lpr printer spoolers are identical.

Section 20 provides a description of lpr printer spooler services and the procedures for configuring those services.

An lpr printer spooler must be installed in each CP or TP to which printers are connected. By default, the MBTOS Config utility installs the small lpr printer spooler manager in the primary partition of the selected processor. It installs the big lpr printer spooler manager in a secondary partition of the selected TP.

The small lpr printer spooler manager run file is

[sys]<sys>MSpoolerMgr.run

The big lpr printer spooler manager run file is

[sys]<sys>MBigSpoolerMgr.run

The big lpr printer spooler manager run statement must be included in the initialization file of any TP to which you want to connect more than four printers (that is, more than one parallel and three serial printers).

When an lpr printer spooler manager in a primary partition is activated on a processor, it does not return control to the CLI to read the next run statement. Therefore, such an lpr printer spooler manager run statement must be the last statement in any initialization file in which it appears. A run statement in an initialization file that appears after such an lpr printer spooler manager run statement will never be executed. This limitation can be overcome by installing the lpr printer spooler into a secondary partition.

When memory is at a premium and no other applications need to be run on the primary application partition, the small lpr printer spooler should be installed into the primary partition.

The small lpr printer spooler manager uses 57 kB of processor memory. The big lpr printer spooler manager uses 150 kB of processor memory.

lp Server

The lp printers connected to an XE 500 processor board (either a CP or TP), or to a PT 1500 connected to a processor (a CP), are controlled by an lp spooler running on that processor. The lp printing queue is controlled by CENTIX system software.

Refer to the *XE 500 CENTIX Administration Guide* for a description of lp printer spooler services and the procedures for configuring those services.

Tape Server

The tape server performs tape handling functions and provides tape services for applications software requiring access to half-inch magnetic tape. The tape server is installed on an SP or DP.

The tape server must be installed in the SP or DP board to which the tape drive is connected.

The tape server's run file, including two optional parameters, is

[sys]<sys>MTapeServer.run,*Drives*,*BuffSize*

where

Drives	is the number of tape drives the server is supporting.
BuffSize	is an optional parameter that specifies the size, in kB, of each buffer allocated. If *BuffSize* is not specified, a default value of 16 is used, meaning a buffer size of 16 kB. The value must be greater than 0 and less than 65. If the default value is used, the resulting buffer size will be 16 kB less 2 bytes; for any other value, the size is *BuffSize* kB.

The tape server uses all available memory on the processor in which it runs.

QIC Server

The QIC server supports the QIC tape drive. The QIC server is based on the tape server, which supports half-inch magnetic tape drives, but with the following differences:

- The QIC server supports only one user at a time.
- The processor on which the QIC server runs must have 768 kB of memory.
- There is only one QIC drive per system.

The selected processor cannot be an SP or DP that is to run a tape server.

The QIC server run file is

[sys]<admin>MQicServer.run

The QIC server run statement appears in a processor initialization file as if it is being installed in a primary partition. However, the QIC server is actually installed in a secondary partition.

The QIC server requires 60 kB of memory.

CENTIX File System Server

The CENTIX file system server acts as an interface between CENTIX I/O requests to a logical disk device and the actual BTOS-based disk I/O operation. The CENTIX file system server makes the software connections between BTOS partitions and the CENTIX file systems that reside in those partitions.

The CENTIX file system server can be installed in a primary or secondary partition of a processor. It is normally installed in a processor that controls disk devices (FP or DP). If the server is running in the master processor, it should be run in a secondary partition. If the server is running in a processor other than the master processor, it can be run in the primary partition.

The CENTIX file system server run statement is

[sys]<sys>UFS.run

The CENTIX file server requires at least 150 kB of memory. If running in a secondary partition, increasing the size of this partition can improve system performance.

Configuring BTOS System Services

You create the standard set of XE 500 BTOS system configuration files by using the MBTOS Config utility. These files define your system's hardware and software configuration.

If you are installing 6.0-release software, refer to the subsection "Creating the BTOS System Configuration Files" later in this section.

Note: *Sections 12 and 13 provide overviews of BTOS system configuration files and system services. You should read these sections before configuring your system with the MBTOS Config utility.*

The MBTOS Config utility provides a menu-driven facility that allows you to modify standard system configuration files in the following ways:

Master Configuration File

- Create the master configuration file if one does not already exist.
- List the contents of the existing master configuration file.
- Add processor board entries to the master configuration file.
- Delete processor board entries from the master configuration file.

Processor Initialization Files

- Create default processor initialization files.
- List the contents of a processor initialization file.
- Add system services to a processor initialization file.
- Delete system services from a processor initialization file.

Processor Configuration Files

- Create default processor configuration files.
- List the contents of a processor configuration file.
- Add or modify entries in a CP or TP configuration file.

- Delete entries from a CP or TP configuration file.
- Add or modify X, Y, or Z block allocation entries in a processor configuration file.

Note: *The entry in a CP configuration file that defines the number of PT 1500s cannot be modified through MBTOS Config. A default number of 1 is entered for the CP00 configuration file when it is created by MBTOS Config. Other CP configuration files get no PT 1500 entries when they are created. These PT 1500 numbers are updated in the appropriate CP configuration file by the centrEASE administrative facility anytime that facility is used to add or remove a PT 1500 to the system.*

When to Use the MBTOS Config Utility

The MBTOS Config utility is used to do the following:

- If you are installing 6.0 release XE 500 BTOS software for the first time, you must use the MBTOS Config utility to create the system configuration files.
- If you have already installed and configured 6.0 release system software and you wish to modify current files or create new files to reflect changes in your system configuration, use the MBTOS Config utility.

Each processor in the system, except APs, must have an initialization file and a configuration file. If you have initialized the system disk as part of the software installation procedure, none of these files currently exist. You must run the MBTOS Config utility to create these files.

The utility allows you to automatically create default versions of all necessary system configuration files.

You can also choose to selectively create nondefault versions of system configuration files. If you do this, make sure that you create an initialization file and a configuration file for each processor in the system.

It is recommended that you first create the default versions for all system configuration files, then run the utility again to modify files as necessary.

Creating the BTOS System Configuration Files

To start the MBTOS Config utility as part of the 6.0 release system software installation, the MBTOS Config utility can only be run by selecting it from the Set Up Software Menu of the restricted mode.

The Set Up Software Menu should be displayed at the terminal upon completion of the CENTIX software installation procedures described in Section 7. Use the following procedure to create the BTOS system configuration files.

1 In response to the Set Up Software Menu prompt, choose MBTOS Config by entering "4" and pressing the RETURN key.

   The MBTOS Config utility begins running. The master configuration file is created and its contents are displayed. Following the master configuration file listing, the Default Menu and prompt are displayed.

Note: *If you need detailed information about using the MBTOS Config utility, refer to the next subsection, "Using the MBTOS Config Utility."*

2 At the Default Menu prompt, enter "2" and press the RETURN key. The utility creates default configuration and initialization files for each processor listed in the master configuration file. It then lists these files.

   After the processor configuration and initialization files are created, "MBTOS Config Utility Complete" is displayed and you are returned to the Set Up Software Menu.

3 Step 2 creates only default versions of the system configuration files.

If any files must be modified to match the actual system configuration, run MBTOS Config again, selecting the "Do Not Use Default" option of the MBTOS Config Default Menu. Refer to the subsections "Selectively Creating or Modifying Processor Initialization Files" and "Selectively Creating or Modifying Processor Configuration Files" later in this section.

If the BTOS system configuration files do not need to be modified, or when you have finished modifying them, go to step 4.

4 In response to the Set Up Software Menu prompt, enter "0" and press the RETURN key.

The Restricted Mode Main Menu is displayed.

5 In response to the Restricted Mode Main Menu prompt, enter "0" and press the RETURN key.

CENTIX begins a modified version of the shutdown utility, during which time the file systems are unmounted. This version of the shutdown utility does not stop active processes.

After about a minute, a message is displayed telling you that you can stop or reset the processor.

6 Turn the keyswitch to STOP and then to NORMAL. This reboots the system in normal mode.

If the system start-up is successful, the STATUS display shows "20" and the following prompt is displayed at the terminal:

```
Is the above date/time correct (y or n)
```

7 Enter "y" and press the RETURN key.

The following prompt is displayed:

```
Do you want to check the file systems? (y or n)
```

8 Enter "y" and press the RETURN key. The system checks the CENTIX root file system, displaying status information about the progress of the checking operation.

When the file system check completes successfully, the following prompt is displayed:

```
login:
```

9 Enter "root" and press the RETURN key. The CENTIX superuser shell prompt, #, is displayed. You are now logged into the CENTIX system as the superuser, which allows you to access all CENTIX files and perform administrative tasks.

Go now to Section 15 for instructions on configuring CENTIX software.

Starting the MBTOS Config Utility

Once 6.0 release system configuration files have been created, the MBTOS Config utility can be invoked by

- Booting up in restricted mode and selecting MBTOS Config from the Set Up Software Menu.
- Invoking the MBTOS Config utility through centrEASE.
- Using the CENTIX **ofcli** command to execute the run file [sys]<admin>MBTOSConfig.run.

Using the MBTOS Config Utility

The MBTOS Config utility is a menu-driven facility through which you create or modify BTOS system configuration files.

Each MBTOS Config menu comprises

- A heading that describes the purpose of the menu.
- A list of actions from which to select. Each action is assigned a number, which is used to enter your selection.

- A prompt at which you enter the number of the action you want to execute. After entering the desired number, you must press the RETURN or GO key to execute the action. (For simplicity, the procedures described later in this section refer only to the RETURN key.)

When this utility is invoked, a welcome message is displayed along with a listing of the standard master configuration file ([sys]<sys>Master.cnf).

If the master configuration file does not exist, it is created dynamically when the utility is invoked. The utility polls the actual processor boards installed in the system to find out what processors there are. It then builds the master configuration file entries, using the standard processor BTOS versions and processor configuration file names.

If the master configuration file already exists, its contents are displayed when the utility is invoked.

The listing of the master configuration file is followed by the main Default Menu. This Default Menu gives you the option to

- Exit the utility and return to the BTOS Executive.
- Automatically create default configuration and initialization files for each processor listed in the master configuration file.
- Modify the master configuration file.
- Selectively create or modify the processor initialization and configuration files.

Go now to the appropriate subsection that follows.

Creating the Default Versions of All Files

The main Default Menu and prompt are displayed as follows:

```
                    DEFAULT MENU

**********************************************
**       0 - EXIT UTILITY                   **
**       1 - DO NOT USE DEFAULT             **
**       2 - USE DEFAULT SYSTEM FILES       **
**********************************************

Please Select Mode of Operation ==>
```

To create the default files, enter "2" and press the RETURN key. The utility creates default configuration and initialization files for each processor listed in the master configuration file. It then lists these files.

Note: *The utility creates the processor configuration and initialization files by copying the contents of general default configuration files that are installed with the standard software.*

There is a default initialization file and default configuration file for each ***type*** *of processor, except the AP. Their file names take the form [sys]<sys>Def*procID*.cnf or [sys]<sys>DefInit*procID*.jcl, where* procID *represents the appropriate processor name and number. They are listed in Appendix B.*

Caution: *These general default configuration files should not be modified or deleted.*

After the processor configuration and initialization files are created, "MBTOS Config Utility Complete" is displayed and the command line is returned to the screen.

Modifying the Master Configuration File

The main Default Menu and prompt are displayed as follows:

```
                    DEFAULT MENU

.............................................
** 0 = EXIT UTILITY                         **
** 1 = DO NOT USE DEFAULT                   **
** 2 = USE DEFAULT SYSTEM FILES             **
.............................................

Please Select Mode of Operation ==>
```

To modify the master configuration file, enter "1" and press the RETURN key. The Main Menu is displayed, from which you can select the type of file you wish to create or modify.

The Main Menu and prompt are displayed as follows:

```
                    MAIN MENU

.............................................
** 0 = EXIT UTILITY                         **
** 1 = MASTER CNF file                      **
** 2 = JCL file                             **
** 3 = CNF file                             **
.............................................

Please Select Type of File ==>
```

To modify the master configuration file, enter "1" and press the RETURN key. A listing of the master configuration file is displayed. This listing is followed by the Function Menu. The Function Menu allows you to choose whether you want to add the entries for a processor board or delete the entries for a processor board.

The Function Menu and prompt are displayed as follows:

```
                    FUNCTION MENU

**********************************************
** 0 - RETURN TO MAIN MENU                   **
** 1 - ADD                                   **
** 2 - DELETE                                **
**********************************************

Please Select a Function for the Master Cnf file ==>
```

Refer to the following two subsections for adding and deleting processor entries from the master configuration file.

Adding Processor Entries

To add the appropriate entries for a processor, enter "1" in response to the Function Menu prompt and press the RETURN key. The Add Menu is displayed, from which you can select the type of processor that you have added to the system.

The Add Menu and prompt are displayed as follows:

```
                      ADD MENU

**********************************************
** 0 - RETURN TO FUNCTION MENU               **
** 1 - CP                                    **
** 2 - DP                                    **
** 3 - FP                                    **
** 4 - SP                                    **
** 5 - TP                                    **
**********************************************

Please Select Processor Board to Add to the Master Cnf ==>
```

Enter the number that corresponds to the type of processor you have added to the system and press the RETURN key. The Add Board Menu and prompt are displayed. The Add Board Menu allows you to select the location in the master configuration file at which the processor entries should appear.

Note: *You cannot configure a secondary AP board into the system through MBTOS Config. You must use centrEASE. Refer to the* XE 500 CENTIX centrEASE Operations Reference Manual.

The Add Board Menu lists each of the processors that have entries in the master configuration file. In between the processor name are numbered locations at which the new processor entries can be added.

Following the Add Board Menu is the prompt

```
Please Select the Location to Add the Processor Board ==>
```

Enter the number that corresponds to the appropriate location and press the RETURN key.

Caution: *The location that you select must correspond to the physical location among the processor boards at which the new processor was added. That is, the sequence of processor entries in the master configuration file must always correspond exactly to the physical sequence of processors in the XE 500 enclosure(s) (that is, by order of slot number).*

After a location has been selected, the updated master configuration file is displayed and the Function Menu is returned to allow further modifications.

When updating the master configuration file, MBTOS Config adds the appropriate entries to the master configuration file. If the processor being added requires an include statement for its configuration file, the include statement is automatically added. Also, the initialization and configuration file names are updated for any processors that follow the new processor and are of the same type. (For example, if a new CP is added to the system between CP01 and CP02, files related to CP02 are automatically renamed to reflect its new number, CP03.)

Note: *You must create the new processor's configuration file, using MBTOS Config.*

Deleting Processor Entries

To delete the appropriate entries for a processor, enter "2" in response to the Function Menu prompt and press the RETURN key. The Delete Board Menu is displayed, from which you can select the processor that you have removed from the system.

The Delete Board Menu lists the processors included in the current master configuration file. Following the Delete Board Menu is the prompt

```
Please Select Processor Board to Delete
from the Master Cnf -->
```

Enter the number that corresponds to the board that has been removed from the system and press the RETURN key. The updated master configuration file is displayed and the Function Menu is returned to allow further modifications.

When updating the master configuration file, MBTOS Config deletes the appropriate entries to the master configuration file. If the processor being deleted had an include statement for its configuration file, the include statement is automatically deleted. Also, the initialization and configuration file names are updated for any processors that followed the processor that was removed and are of the same type. (For example, if the system had CP01 and CP02, and CP01 was removed, files related to CP02 are automatically renamed to reflect its new number, CP01.)

Selectively Creating or Modifying Processor Initialization Files

Note: *Before attempting to modify processor initialization files, you must know restrictions as to where system services can be run, how many copies of a given system service can be run, and so on. Refer to "Guidelines for Running System Services" later in this section.*

The main Default Menu and prompt are displayed as follows:

```
                    DEFAULT MENU

.................................................
** 0 = EXIT UTILITY                             **
** 1 = DO NOT USE DEFAULT                       **
** 2 = USE DEFAULT SYSTEM FILES                 **
.................................................

Please Select Mode of Operation ==>
```

To create or modify a processor initialization file, enter "1" and press the RETURN key. The Main Menu is displayed, from which you can select the type of file you wish to create or modify.

The Main Menu and prompt are displayed as follows:

```
                     MAIN MENU

.................................................
** 0 = EXIT UTILITY                             **
** 1 = MASTER CNF file                          **
** 2 = JCL file                                 **
** 3 = CNF file                                 **
.................................................

Please select Type of file ==>
```

Note: *The MBTOS Config utility refers to processor initialization files as "JCL" files.*

To modify a processor initialization file, enter "2" and press the RETURN key.

The Board Menu is displayed. The Board Menu allows you to choose which processor's initialization file you want to create or modify. The utility examines the master configuration file to generate the list of processor boards.

Following the Board Menu is the prompt:

```
Please Select a Specific JCL file ==>
```

Enter the number corresponding to the processor board whose initialization file you want to create or modify. Then press the RETURN key. The JCL Default Menu appears. This menu allows you to

- Create a default initialization file for the processor.
- Create your own initialization file for the processor.
- Modify the processor's existing initialization file.

The JCL Default Menu and prompt are displayed as follows:

```
                    DEFAULT MENU

**********************************************
** 0 - RETURN TO MAIN MENU                   **
** 1 - DO NOT USE DEFAULT                    **
** 2 - USE DEFAULT JCL FILE                  **
**********************************************

Please Select Mode of Operation ==>
```

Creating the Default Version

To create the default version of the processor initialization file, enter "2" and press the RETURN key. The default initialization file is created and displayed.

Note: *The utility copies the contents of the general default initialization file corresponding to the processor board type into the processor initialization file. If the processor initialization file does not already exist, a new file is opened before the copy is made. If it does exist, it is overwritten during the copy operation.*

After the new initialization file is displayed, the utility returns to the Main Menu.

Modifying an Existing File or Creating a Nondefault File

To modify the existing processor initialization file or to create a nondefault one, enter "1" in response to the JCL Default Menu prompt and press the RETURN key. If the file already exists, its contents are displayed. If it is a new file, an empty file results.

The listing of the initialization file is followed by a Function Menu. The Function Menu allows you to choose whether you want to add or delete a system service entry.

The Function Menu and prompt are displayed as follows:

```
                FUNCTION MENU

**********************************************
** 0 = RETURN TO MAIN MENU                   **
** 1 = ADD                                   **
** 2 = DELETE                                **
**********************************************

Please Select a Function for the JCL file ==>
```

To add a system service to the file, enter "1" and press the RETURN key. An Add Server Menu and prompt are displayed. This menu lists the system services that can be run on the processor. The Add Server Menu is followed by the prompt

```
Please Select a Server to Add to the JCL file ==>
```

Enter the number corresponding to the service you wish to add and press the RETURN key. The updated file is displayed and the Function Menu and prompt are displayed again for further modifications.

To delete a system service from the file, enter "2" and press the RETURN key. A Delete Server Menu and prompt are displayed. This menu lists the system services that are currently to be run on the processor. The Delete Server Menu is followed by the prompt

```
Please Select a Server to Delete from the JCL file ==>
```

Enter the number corresponding to the service you wish to delete and press the RETURN key. The updated file is displayed and the Function Menu and prompt are displayed again for further modifications.

When you are finished modifying the file, enter "0" in response to the Function Menu prompt to return to the Main Menu.

Selectively Creating or Modifying Processor Configuration Files

The main Default Menu and prompt are displayed as follows:

```
                    DEFAULT MENU

**********************************************
** 0 - EXIT UTILITY                          **
** 1 - DO NOT USE DEFAULT                    **
** 2 - USE DEFAULT SYSTEM FILES              **
**********************************************

Please Select Mode of Operation ==>
```

To create or modify a processor configuration file, enter "1" and press the RETURN key. The Main Menu is displayed, from which you can select the type of file you wish to create or modify.

The Main Menu and prompt are displayed as follows:

```
                    MAIN MENU

.............................................
** 0 - EXIT UTILITY                         **
** 1 - MASTER CNF file                      **
** 2 - JCL file                             **
** 3 - CNF file                             **
.............................................

Please select Type of file ==>
```

Note: *The MBTOS Config utility refers to processor configuration files as "CNF" files.*

To modify a processor configuration file, enter "3" and press the RETURN key.

The Board Menu is displayed. The Board Menu allows you to choose which processor's configuration file you want to create or modify. The utility examines the master configuration file to generate the list of processor boards.

Following the Board Menu is the prompt:

```
Please Select a Specific CNF file ==>
```

Enter the number corresponding to the processor board whose configuration file you want to create or modify. Then press the RETURN key. The CNF Default Menu appears. This menu allows you to

- Create a default configuration file for the processor.
- Create your own configuration file for the processor.
- Modify the processor's existing configuration file.

Note: *Currently, the MBTOS Config utility can only add or modify Y or Z block allocation entries in FP, DP, or SP configuration files. To selectively create an FP, DP, or SP configuration file, you must use the default version of the file.*

If you must modify entries in an FP, DP, or SP configuration file other than Y or Z block entries, use the ofed or ofvi editor. Follow the guidelines provided in Section 12 of this manual.

The CNF Default Menu and prompt are displayed as follows:

```
                DEFAULT MENU

**********************************************
** 0 - RETURN TO MAIN MENU                   **
** 1 - DO NOT USE DEFAULT                    **
** 2 - USE DEFAULT CNF FILE                  **
**********************************************

Please Select Mode of Operation ==>
```

Creating the Default Version

To create the default version of the processor configuration file, enter "2" and press the RETURN key. The default configuration file is created and displayed.

Note: *The utility copies the contents of the general default configuration file corresponding to the processor board type into the processor configuration file. If the processor configuration file does not already exist, a new file is opened before the copy is made. If it does exist, it is overwritten during the copy operation.*

After the new configuration file is displayed, the utility returns to the Main Menu.

Modifying an Existing File or Creating a Nondefault File

To modify an existing TP or CP processor configuration file or to create a nondefault one, enter "1" in response to the CNF Default Menu prompt and press the RETURN key. If the file already exists, its contents are displayed. If it is a new file, an empty file results.

The listing of the configuration file is followed by a Function Menu. The Function Menu allows you to choose whether you want to add or delete a port definition entry.

The Function Menu and prompt are displayed as follows:

```
                FUNCTION MENU

.............................................
** 0 - RETURN TO MAIN MENU                  **
** 1 - ADD                                  **
** 2 - DELETE                               **
.............................................

Please Select a Function for the CNF file ==>
```

The following subsections describe the procedures for adding or deleting entries to TP and CP configuration files

Note: *Do not use MBTOS Config to add or delete terminal port entries. These entries are added or deleted automatically when you use centrEASE to install or remove a terminal.*

Caution: *A CP configuration file must have at least one async entry for the CP to boot.*

Adding a TP RS-232-C Port Entry

Note: *If you select a port entry that is already in the file, you are told that you have selected a duplicate entry and to try again. To modify an existing port entry, you must first delete the entry from the file and then add the new version of it.*

To add an RS-232-C port entry to a TP's configuration file, enter "1" in response to the Function Menu and press the RETURN key. A prompt is displayed that allows you select the port entry you wish to add or to return to the Function Menu. The prompt is displayed as follows:

```
* Valid answers are numbers between 1 and 10 *
Press 0 to Return or Insert a Port Number to Configure ==>
```

Enter the number corresponding to the RS-232-C port entry you wish to add, and press the RETURN key. The following is displayed:

```
The Current Line Being Added
async n, speed=1200, parity=none, stopbits=1, charbits=8
```

where *n* is the port number that you selected. This display of the port entry is followed by a Change Line Menu.

The Change Line Menu allows you to

- Include the selected port entry in the TP configuration file.
- Cancel the addition of the selected port entry and return to the Port Menu to select another port.
- Change the line speed defined in the selected port entry.

The Change Line Menu and prompt are displayed as follows:

```
                CHANGE LINE MENU

.........................................
** 0 = RETURN TO ADD MENU               **
** 1 = PORT                             **
** 2 = SPEED                            **
.........................................

Please Select the Parameter to Change ==>
```

To add the port entry, as displayed, to the TP configuration file, enter "0" and press the RETURN key. The entry is added, the updated TP configuration file is displayed, and the Add Menu and prompt are displayed again to allow further modifications.

To cancel the addition of the port entry, enter "1" and press the RETURN key. The entry is not added and the port selection prompt is displayed again to allow you to select another port.

To change the line speed defined in the port entry, enter "2" and press the RETURN key. The Speed Menu and prompt are displayed. This menu allows you to set the line speed for the RS-232-C port. The values provided are in baud. The Speed Menu also allows you to return to the Change Line Menu.

The Speed Menu is followed by the prompt

```
Please Select a Speed ==>
```

Enter the number corresponding to the desired speed and press the RETURN key. The modified port entry is displayed, followed by the Change Line Menu. The port entry is not actually added to the TP configuration file until you return to the Add Menu.

Adding a CP RS-232-C Port Entry

Note: *If you select a port entry that is already in the file, you are told that you have selected a duplicate entry and to try again. To modify an existing port entry, you must first delete the entry from the file and then add the new version of it.*

To add an RS-232-C port entry to a CP's configuration file, enter "1" in response to the Function Menu and press the RETURN key. An Add Menu is displayed. This menu lists the CP's port types. The Add Menu is followed by the prompt

```
Please Select an Item to Configure ==>
```

Enter the number corresponding to RS-232-C and press the RETURN key. A Port Menu is displayed, listing the CP's RS-232-C ports.

The Port Menu is followed by the prompt

```
Please Select a Port to Configure ==>
```

Enter the number corresponding to the port entry you wish to add or modify and press the RETURN key. The following is displayed:

```
The Current Line Being Added
async n, speed=1200, parity=none, stopbits=1, charbits=8
```

where *n* is the port number that you selected. This display of the port entry is followed by a Change Line Menu.

The Change Line Menu allows you to

- Include the selected port entry in the CP configuration file.
- Cancel the addition of the selected port entry and return to the Port Menu to select another port.
- Change the line speed defined in the selected port entry.

The Change Line Menu and prompt are displayed as follows:

```
                CHANGE LINE MENU

****************************************
** 0 = RETURN TO ADD MENU              **
** 1 = PORT                            **
** 2 = SPEED                           **
****************************************

Please Select the Parameter to Change ==>
```

To add the port entry, as displayed, to the CP configuration file, enter "0" and press the RETURN key. The entry is added, the updated CP configuration file is displayed, and the Add Menu and prompt are displayed again to allow further modifications.

To cancel the addition of the port entry, enter "1" and press the RETURN key. The entry is not added and the Port Menu and prompt are displayed again to allow you to select another port.

To change the line speed defined in the port entry, enter "2" and press the RETURN key. The Speed Menu and prompt are displayed. This menu allows you to set the line speed for the RS-232-C port. The values provided are in baud. The Speed Menu also allows you to return to the Change Line Menu.

The Speed Menu is followed by the prompt

```
Please Select a Speed ==>
```

Enter the number corresponding to the desired speed and press the RETURN key. The modified port entry is displayed, followed by the Change Line Menu. The port entry is not actually added to the CP configuration file until you return to the Add Menu.

Adding a CP RS-422 Cluster Line Entry

Note: If you select a line entry that is already in the file, you are told that you have selected a duplicate entry and to try again. To modify an existing line entry, you must first delete the entry from the file and then add the new version of it.

To add an RS-422 cluster line entry to a CP's configuration file, enter "1" in response to the Function Menu and press the RETURN key. An Add Menu is displayed. This menu lists the CP's port types. The Add Menu is followed by the prompt

```
Please Select an Item to Configure ==>
```

Enter the number corresponding to RS-422 and press the RETURN key. A Line Menu is displayed, listing the processor's RS-422 lines.

The Line Menu is followed by the prompt

```
Please Select a Line to Configure ==>
```

Enter the number corresponding to the line entry you wish to add or modify and press the RETURN key. The following is displayed:

```
The Current Line Being Added
line n, clock=307k
```

where *n* is the line number that you selected. This display of the line entry is followed by a Change Line Menu.

The Change Line Menu allows you to

- Include the selected line entry in the CP configuration file.
- Cancel the addition of the selected line entry and return to the Port Menu to select another line.
- Change the line speed defined in the selected line entry.

The Change Line Menu and prompt are displayed as follows:

```
                CHANGE LINE MENU

.........................................
** 0 - RETURN TO ADD MENU               **
** 1 - LINE                             **
** 2 - SPEED                            **
.........................................

Please Select the Parameter to Change ==>
```

To add the line entry, as displayed, to the CP configuration file, enter "0" and press the RETURN key. The entry is added, the updated CP configuration file is displayed, and the Add Menu and prompt are displayed again to allow further modifications.

To cancel the addition of the line entry, enter "1" and press the RETURN key. The entry is not added and the Port Menu is displayed again to allow you to select another line.

To change the line speed defined in the line entry, enter "2" and press the RETURN key. The Speed Menu is displayed. This menu allows you to set the line speed for the RS-422 line. The values provided are in baud. The Speed Menu also allows you to return to the Change Line Menu.

The Speed Menu is followed by the prompt

```
Please Select a Speed ==>
```

Enter the number corresponding to the desired speed and press the RETURN key. The modified line entry is displayed, followed by the Change Line Menu. The line entry is not actually added to the CP configuration file until you return to the Add Menu.

Note: *PT 1500s can only be run at 307 k baud.*

Deleting A Port or Block Entry

To delete an entry from a CP or TP configuration file, enter "2" in response to the Function Menu and press the RETURN key. A Delete Line Menu is displayed. This menu lists the entries that are currently in the processor configuration file. It also lists an option that allows you to return to the Function Menu. The Delete Line Menu is followed by the prompt

```
Please Select a Line to Delete from the CNF file ==>
```

Enter the number corresponding to the entry you wish to delete and press the RETURN key. The updated file is displayed and the Function Menu is displayed again for further modifications.

When you are finished modifying the file, enter "0" in response to the Function Menu prompt to return to the Main Menu.

Adding or Modifying an X, Y, or Z Block Entry

To add or modify an X, Y, or Z block entry to a processor's configuration file, enter "1" in response to the CNF Default Menu prompt and press the RETURN key. If the file already exists, its contents are displayed. If it is a new file, an empty file results.

The listing of the configuration file is followed by a Function Menu. The Function Menu allows you to choose whether you want to add or delete an entry.

The Function Menu and prompt are displayed as follows:

```
                    FUNCTION MENU

**********************************************
** 0 - RETURN TO MAIN MENU                   **
** 1 - ADD                                   **
** 2 - DELETE                                **
**********************************************

Please Select a Function for the CNF file ==>
```

Enter "1" in response to the Function Menu and press the RETURN key. An Add Menu is displayed. The Add Menu lists the types of entries that can be added. The Add Menu is followed by the prompt

```
Please Select an Item to Configure ==>
```

Enter the number corresponding to an X, Y, or Z block entry and press the RETURN key. A Block Menu and prompt are displayed. This menu lists the different blocks you can add or modify. This menu is followed by the prompt

```
Please Select a Block to Configure ==>
```

Enter the number corresponding to the block you wish to add or modify. If a block entry exists, the following message appears:

```
The Current Line Being Added

BlockType=n, Number=xxx


* Valid answers are numbers between 000 and 200 *

Press 000 to Return or Insert the Number
of Blocks to Configure ==>
```

where *n* is the block type you selected and *xxx* is the current number already configured.

If that block entry does not exist, then the first two lines of the previous messages do not appear.

Enter the desired number of blocks for the chosen block type and press the RETURN key. You must enter a three-digit number (for example, to specify 12 blocks, enter "012"). The utility adds or updates the current block entry to the configuration file as specified, displays the new block entry, and returns to the Block Menu.

To leave the current block entry unchanged, enter "000". The utility returns to the Block Menu.

Guidelines for Running System Services

The MBTOS Config utility allows you to install BTOS system services on specific processor boards. Table 14-1 lists the available system services, their corresponding run file names, and the amount of memory they require.

Table 14-1 **XE 500 BTOS System Services and Their Run Files**

System Service	Run File	Memory
AdminAgent, single-mode	[sys]<sys>MFAdminAgent.run	64 kB
Master AdminAgent, multiple mode	[sys]<sys>SysAdminAgent.run	32 kB
Slave AdminAgent, multiple mode	[sys]<sys>SlvAdminAgent.run	56 kB
Queue Manager	[sys]<sys>MInstallQMgr.run	16 kB minimum
Small lpr Printer Spooler Manager	[sys]<sys>MSpoolerMgr.run	57 kB
Big lpr Printer Spooler Manager	[sys]<sys>MBigSpoolerMgr.run	150 kB
lp Server	[sys]<sys>LpServer.run	7 kB
Tape Server	[sys]<admin>MTapeServer.run	All available memory on processor
QIC Server	[sys]<admin>MQicServer.run	60 kB
CENTIX File System Server	[sys]<sys>UFS.run	150 kB

Restrictions apply as to how MBTOS Config handles adding system services, such as on which processors they can run, the number of copies of a given system service that can be run in the system, and whether they are to be run in a primary or secondary partition. Table 14-2 lists the system services and applicable restrictions.

Table 14-2 **Restrictions for XE 500 BTOS System Services**

System Service	Can Run on:	Number	Partition
AdminAgent, single-mode	Any	One per processor	Primary
Master AdminAgent, multiple-mode	Any	One per system	Secondary
Slave AdminAgent, multiple-mode	Any	One per processor	Primary
Queue Manager	Any	One per system	Primary
Small Printer Spooler Manager	CP, TP	One per CP or TP	Primary
Big Printer Spooler Manager	TP	One per TP	Secondary
Ip Server	CP, TP	One per CP or TP	Primary
Tape Server	DP or SP	One per system	Primary
QIC Server	FP	One per system	Secondary
CENTIX File System Server	FP, DP	One per system	Secondary

In addition to those listed in Table 14-2, the following restrictions apply:

- The run statement for the single-mode AdminAgent, slave AdminAgent, small lpr printer spooler manager, lp Server, and Tape Server must be the last line of the file in which they appear.
- Either a small lpr printer spooler manager or a big lpr printer spooler manager can be run on a TP, but not both.
- If you receive the BTOS error code 400 (out of processor memory), use the BTOS MPLog utility to list the contents of [sys]<sys>log.sys. If the log states that the 400 error occurred because too many servers are installed on the processor, you should make sure the processor has the maximum amount of memory (786 kB) or move system services to another processor.

MBTOS Config "-Old" Files

The MBTOS Config utility does not keep a log file of each session. However, for any configuration files that are modified, the original file is saved and "-old" is appended to the file name. This allows you to keep the previous version of the file. The "old" version of the file may be needed if the modified version causes system problems, or is lost or corrupted.

Note: *Only the previous generation of a configuration file is saved as an "old" file.*

For example, assume that you had modified the original CP00 configuration file. If you modify this file again, the original file, now appended with "-old," is overwritten with the first modified version of the file.

Rebooting the System After Running MBTOS Config

In order to implement any modifications made to system configuration files, you must reboot the system by turning the keyswitch to STOP and then to NORMAL. If you do not reboot the system, the system continues to run based on the system configuration defined by the configuration files before they were modified.

Configuring CENTIX Software

After configuring the BTOS system files, it is recommended that you perform the following tasks related to configuring CENTIX software:

- Add additional APs to the system.
- Create file systems.
- Add CENTIX users to the system.
- Add terminals to the system.
- Install your CENTIX environmental software products.

You may also want to

- Configure the lpr printer spooler service from the default configuration.
- Configure the lp printer spooler service into the system.
- Change the terminal to which your system console is assigned.
- Configure a remote PT 1500.

Some of these tasks are described in this section. Others are mentioned, but you are referred to the appropriate sections in this guide for the related procedures.

When you have completed configuring the CENTIX software, go to Section 16 and perform the applicable post-configuration tasks.

Adding Application Processors

The system, as installed, only supports one AP (AP00). If you have other APs in your system, they must be added to the system through the Reconfiguring the System function of centrEASE.

Refer to the *XE 500 CENTIX centrEASE Operations Reference Manual* for the procedure for running the Reconfiguring the System function of centrEASE.

Creating CENTIX File Systems

The system, as installed, contains only those file systems related to the CENTIX root. All other file systems must be added to the system through the Managing CENTIX File Systems function of centrEASE.

Refer to the *XE 500 CENTIX centrEASE Operations Reference Manual* for the procedure for running the Managing CENTIX File Systems function of centrEASE.

Adding CENTIX Users

The system, as installed, only supports users logging in as "root." All other users must be added to the system through the Managing User Accounts function of centrEASE.

Refer to the *XE 500 CENTIX centrEASE Operations Reference Manual* for the procedure for running the Managing User Accounts function of centrEASE.

Adding Terminals

The system, as installed, only supports one terminal. All other terminals must be added to the system through the Reconfiguring the System function of centrEASE.

Refer to the *XE 500 CENTIX centrEASE Operations Reference Manual* for the procedure for running the Reconfiguring the System function of centrEASE.

Installing CENTIX Environmental Software

Refer to Section 9 for instructions on installing CENTIX environmental software.

Configuring Printer Spooler Services

The lpr printer spooler service is installed with the system software. It does not require any configuration if you wish to use the default version of the service. Refer to Section 20 for a complete description of the lpr service and how to reconfigure it.

The software for the lp printer service is installed with the system. However, to use this service you must configure it into the CENTIX system. Refer to the *XE 500 CENTIX Administration Guide* for a complete description of the lp service and how to configure it.

Changing the System Console

The CENTIX system reports all system status messages to the system console. By default for systems with PT 1500s, the system console is the first PT 1500 connected to CP00 that receives a getty (that is, that boots up).

You can change the terminal to which the system console is assigned. Refer to the *XE 500 CENTIX Administration Guide* for details.

Configuring a Remote PT 1500

A remote PT 1500 is a PT 1500 that is connected to the XE 500 via an RS-232-C serial line rather than the conventional RS-422 cluster line.

To configure a remote PT 1500 into CENTIX, use the following procedure:

1 Using the Add a Terminal function of centrEASE, add the remote PT 1500 as an RS-232-C terminal. Note the tty number that centrEASE assigns to the terminal.

2 If necessary, change the line speed (from the default 9600 baud) in the line that centrEASE added for the terminal in the /etc/inittab*nn* file, where *nn* is the number of the AP to which the terminal is assigned.

3 If the remote PT 1500 is to be used as a console, add the parameters "flowgen, flowact" to the async line entry in the configuration file of the CP or TP to which the PT 1500 is connected.

4 If the PT 1500 is connected through a modem, add the entry ",modem" to the async line entry in the configuration file of the CP or TP to which the PT 1500 is connected.

5 For any user of the remote PT 1500, add the following to the user's .profile file:

```
TERM=pt
export TERM
```

6 Make the following changes to the /etc/gettydefs file:

a Add the RS-232-C line speed, if it is not listed.

b Add IXOFF to the final flags.

c Remove IXANY from the final flags.

Post-Configuration Tasks

After creating the system configuration files, it is recommended that you perform the following tasks:

- Restore any customized BTOS and CENTIX system files that you saved prior to the software installation.
- Get a listing of the CENTIX files installed.
- Create crash dump files for processor boards.
- Get a file and printed listing of the bad spots for all disks in your system.
- Backup your BTOS and CENTIX system software.
- Duplicate any software release disk cartridges.

This section contains descriptions and procedures for these tasks.

Restoring Customized System Files

You can now restore any customized BTOS system files or CENTIX files that you had saved before beginning the software installation procedure.

It is recommended that you use the backup function of centrEASE to backup any 6.0 release system files that will be overwritten when you restore your old customized files. Then, if you do decide to use a 6.0 release system file, it is easier to use the centrEASE restore function on the archived copy than to run MSysLoad to bring the file back into the system. For information on the centrEASE backup and restore functions, refer to the *XE 500 CENTIX centrEASE Operations Reference Manual*.

Listing CENTIX Files

You may want to get a listing of all of the CENTIX files just installed. To do this, enter the following command as superuser:

```
# find / -depth -print
```

This command causes a listing of all CENTIX files at your terminal. You may want to pipe this listing to a printer to get a hard copy of it, or pipe it through the **more** command so that you can control the scrolling of the list at the terminal screen.

Creating BTOS Processor Crash Dump Files

When a BTOS processor detects a fatal system error, it dumps its memory contents to its crash dump file, if one exists. This file can then be examined to determine the cause of the problem. (Refer to the *XE 500 CENTIX Administration Guide* for more information about crash dump files.)

The crash dump file for the master processor is created when the BTOS system disk is initialized. However, crash dump files for the other BTOS processors must be created manually.

If you wish to create a crash dump file for a processor, enter the following CENTIX command as superuser:

crup '[sys]<sys>*Xpnn*.crash' *m*

where *Xpnn* is the processor board (for example, CP01, SP00) and *m* is the size, in blocks, of the total onboard memory for the processor, including expansion memory (1 block = 512 bytes).

Recording XE 500 Disk Bad Spots

If you ever have to reinitialize an XE 500 disk, you will need a listing of the disk's bad spots.

Use the BTOS MVolume Report utility to list the information about the bad spots on each disk. Use the "[Print file]" option of the utility to copy this information to a printable file. Print the file. Save the hardcopy and file listing of each disk's bad spots in a safe place.

See the *XE 500 BTOS Operations Reference Manual* for information about the MVolume Report utility. See *XE 500 CENTIX System Administration Guide* for instructions on how to execute BTOS utilities from CENTIX and for information about bad spots.

Backing Up System Software

To protect yourself in the event that system software becomes corrupted, you should backup your system software. Then, if system software does become corrupted, you can bring the system up in the restricted mode and restore the corrupted system files from the archive media. This will save you from having to reinstall the corrupted system files from release media.

System software backup operations are handled through the Backing Up and Restoring Data function of centrEASE.

The applicable procedures for backing up and restoring your BTOS and CENTIX system software are described in the *XE 500 CENTIX centrEASE Operations Reference Manual.*

Duplicating Software Release Disk Cartridges

Note: *There is currently no utility to duplicate release software distributed on tape and preserve the original format. It is therefore strongly recommended that you store the release tapes in a safe place.*

To protect yourself in the event that your original release disk cartridges become corrupted, you should duplicate any software release disk cartridges. The procedure for duplicating release disk cartridges copies the cartridge contents to a temporary file on a built-in disk or SMD disk. It then copies them back to another disk cartridge and removes the temporary file.

To duplicate a release disk cartridge, use the following procedure:

1 Logged in as superuser, execute the **ofcli** command by entering "ofcli" after the superuser prompt (#) and pressing the RETURN key.

  The system tells you that it is entering the administrative mode and displays a dollar sign ($) prompt.

2 After the dollar sign prompt, enter

  ```
  $call [d0]<sys>CopyRelease.jcl,[volname]<dirname>
  ```

  and press the RETURN key. The designations *volname* and *dirname* are the BTOS volume and directory names where the temporary copy of the disk cartridge contents is to be stored. The volume must have enough free space to store the temporary copy (a disk cartridge can store up to 5 MB of data). The directory must already exist.

  The system begins copying the disk cartridge contents.

3 After the system has finished copying the contents of the release cartridge onto an XE 500 disk, it displays a message asking you to remove the release cartridge and to insert the cartridge onto which you want the system software copied.

  Remove the original release cartridge. Insert a blank, initialized disk cartridge and press the GO key.

4 The system begins copying the temporary copy to the disk cartridge contents. The system informs you when the copy operation has completed.

   Remove the duplicate release cartridge, and store both the original and the duplicate in a safe place.

5 For other release cartridges in a set, repeat steps 2 through 4.

Note: *A copy of a release cartridge can be produced from a previously produced copy.*

To create a special directory for the temporary file, use the BTOS MCreate Directory utility before you try to duplicate any release cartridges.

Note that this operation copies the contents of one cartridge to an XE 500 disk, and then onto another cartridge, thus creating a duplicate cartridge. This operation uses archive file procedures to create the duplicate release cartridge.

You should use the duplicated version of the system software to reinstall the software if a subsequent recovery from an XE 500 disk does not work. You should use the Burroughs-distributed cartridges only if the duplicate release cartridges fail. Ideally, the originals will never have to be reused.

Configuring the AdminAgent

AdminAgent is the system service that manages the execution of BTOS utilities on XE 500 processors. AdminAgent can run in one of two modes—single or multiple.

Note: *To run an AdminAgent on a processor, the processor must have 768 kB of memory.*

Single Mode

In single mode, the AdminAgent is run in the primary partition of the processor whose initialization file contains the AdminAgent run statement. The run file for single mode AdminAgent is

[sys]<sys>MfAdminAgent.run

In single mode, only one processor can be running AdminAgent. Also, only one user can be executing BTOS utilities at a time.

If you want to run single-mode AdminAgent, include the run statement for [sys]<sys>MfAdminAgent.run in a processor's initialization file.

Note: *Though it is not recommended, the single mode AdminAgent can be installed in more than one processor. The first one to serve a particular request will prevent the others from executing. Therefore, if more than one single mode AdminAgent is installed, you cannot know which one is executing a given BTOS utility.*

Multiple Mode

In multiple mode, several AdminAgents can be running at one time. One of the AdminAgents serves as the master AdminAgent, while the others are slave AdminAgents.

Running multiple AdminAgents allows you to direct that a BTOS utility run on a preferred type of processor or even a specific processor. You can also have the system prompt the user for the specific processor on which a BTOS utility is to run when its master command is executed. These options allow you to enhance system performance—a BTOS utility can be run on the processor type that can best execute the utility. For example, tape-related BTOS utilities can be specified to run through an SP or DP AdminAgent, where they will run faster than on other types of processors.

Running multiple AdminAgents is transparent to users.

The master AdminAgent receives the requests to execute BTOS utilities and directs them to a slave AdminAgent. The slave AdminAgent then executes the utility just as a single mode AdminAgent would. If all slave AdminAgents are busy, the master executes the utility itself.

Multiple AdminAgent Run Statements

The run file for the master AdminAgent is

```
[sys]<sys>sysAdminAgent.run
```

The master AdminAgent should be run in a secondary partition with a size of at least 32 kB.

The run file for a slave AdminAgent is

```
[sys]<sys>slvAdminAgent.run
```

The following rules apply to slave AdminAgents:

- There must be at least one slave AdminAgent installed.
- The slave must be installed in the processor's primary partition.

- The slave can be installed in any BTOS processor, including the processor that is running the master AdminAgent in a secondary partition.
- The slave AdminAgent run statement must be the last entry of the initialization file because the slave AdminAgent run file does not return control to the CLI.
- Allow at least 56 kB of a primary partition for running a slave AdminAgent.

Caution: *When running multiple AdminAgents, the single-mode AdminAgent is not used. Therefore, if you are going from single to multiple AdminAgent mode, the run statement for the single-mode AdminAgent must be removed from the processor initialization files in which it is listed.*

Executing Utilities on Preferred Processors

By executing the **ofcli** command with the **-p** flag, you can specify preferred processors for running certain BTOS utilities. This option allows you to enhance system performance because a BTOS utility can be run on the processor type that can best execute the utility. For example, tape-related BTOS utilities can be specified to run through a SP or DP AdminAgent, where they will run faster than on other types of processors.

To specify a preferred processor or processor type for running a BTOS utility, use the following **ofcli** command format:

```
# ofcli -p Xp[nn]
```

where *Xp* is the processor type designation (for example, FP, CP) and *nn* is the processor number. Do not include the processor number if you are specifying only a preferred processor type.

A slave AdminAgent must be running on the processor designated in the **ofcli** command line.

When preferred processors have been specified, the master AdminAgent attempts to dispatch a utility request using the following hierarchy:

1 The designated processor.

2 A processor of the same type as the designated processor.

3 The least recently used of the available slave AdminAgents.

Note that a preferred and not a required processor is specified by the **-p** flag of the **ofcli** command. This means that a BTOS utility is not prevented from being executed merely because a preferred processor is not available.

Executing Utilities on a Required Processor

By executing the **ofcli** command with the **-f** flag, you can specify a required processor for running a BTOS utility. This option allows you to enhance system performance because a BTOS utility can be run on the processor type that can best execute the utility. For example, tape-related BTOS utilities can be specified to run through a SP or DP AdminAgent, where they will run faster than on other types of processors.

To specify a required processor for running a BTOS utility, use the following **ofcli** command format:

```
# ofcli -f Xpnn
```

where *Xpnn* is the processor designation (for example, FP00, CP01).

A slave AdminAgent must be running on the processor designated in the **ofcli** command line.

When a required processor has been specified, the master AdminAgent attempts to dispatch a utility request to the slave AdminAgent on that processor. If that processor's AdminAgent is busy, or that processor is does not have a slave AdminAgent, the system tells you that the utility could not be executed.

For more information about using the **ofcli** command for executing BTOS utilities from CENTIX, refer to the *XE 500 CENTIX Administration Guide*. For more information about BTOS utilities in general, refer to the *XE 500 BTOS Operations Reference Manual*.

Getting AdminAgent Status Information

To get status information about the AdminAgents in the system, use the following **ofcli** command format:

```
# ofcli -s
```

For a single-mode AdminAgent, the status information indicates the processor on which the AdminAgent is running.

For a multiple-mode AdminAgent, status information includes a list of processors on which slave AdminAgents are running and their current status.

Caution: *Do not attempt to obtain AdminAgent status by executing the BTOS MAdminAgent Status utility through* ***ofcli****.*

Configuring the BTOS Queue Manager

The BTOS Queue Manager controls the various BTOS queues in which related tasks are stored for processing. For example, the Queue Manager controls the queues that are used to handle print requests made through lpr printer spoolers. It can also manage similar queues for data communications processes and user-defined processes.

Queue Index File

The BTOS queue index file, [sys]<sys>Queue.Index, is a text file that contains one entry for each queue under the control of the Queue Manager. A queue entry defines:

1 The queue's name (to be used by the system).

2 The queue entry file to hold the queue's requests.

3 The size of each queue entry.

4 The queue type.

Each queue entry has the following format:

QueueName/FileSpec/EntrySize/QueueType

where

QueueName	is a name unique to your installation; it can be any character string except device names such as comm, kbd, lpt, and so on.
FileSpec	is the name of the queue entry file in which entries from client processes are stored. This file name can be any name you choose; it is recommended that it somehow reflect the name of the queue it serves.
EntrySize	is the size of an entry (number of sectors). The first 40 bytes are reserved for the Queue Manager.

QueueType is the type of queue, numbered according to the following table (Burroughs reserves type 0-80; values may range from 0-255):

Queue Type	Assignment
1	Printer spooler queue
2	BSC RJE queue
3	Batch queue
4	SNA RJE queue

Installing the Queue Manager

The Queue Manager can be installed into a processor by

- Executing the MInstall Queue Manager utility through the **ofcli** command, using the **-f** flag option to force it to run on a designated processor. For more information about running a BTOS utility on a required processor, refer to Section 17.
- Including its run statement in a BTOS processor's initialization file. This causes the Queue Manager to be installed automatically at system start-up.

The following rules apply for running the Queue Manager:

- There can only be one XE 500 Queue Manager installed and running at a time.
- It can run on any BTOS processor.
- The Queue Manager, as released with 6.0 system software, uses 16 kB of processor memory. If caching is used or if the number of queues exceeds the default number, it will use more than 16 kB.

To have the Queue Manager installed automatically at system start-up, use the MBtos Config utility to add the MInstall Queue Manager run statement in the desired processor's initialization file.

The MInstall Queue Manager utility run file has the form

[sys]<sys>MInstallQMgr.run,*cache*,*queuenumber*

where

cache is "yes" or "no". Enter "yes" to have the Queue Manager installed with an extra buffer area. This extra buffer area is used for caching disk access of queue entries. Each cache entry, one for each queue, uses approximately 512 bytes of memory. The cache is meant to increase the speed of the Queue Manager.

To not have the Queue Manager installed with an extra buffer area, enter "no" or leave this field blank.

queuenumber is the maximum number of queues that the manager is to handle. If the number of queue entries in the queue index file exceeds the number specified here, the number of queue entries is used as the maximum number of queues to be handled.

To use the default value of 20, leave this field blank.

This parameter is used to allocate buffer space for the queues.

The default parameters are used when the Queue Manager is installed at system start-up, unless you manually edit the run statement to include nondefault parameters.

Stopping the Queue Manager

To stop the Queue Manager from running, regardless of how it was installed, execute the MDeInstall Queue Manager utility through the **ofcli** command. Use the **-f** flag option to force the utility to run on a designated processor. For more information about running a BTOS utility on a required processor, refer to Section 17.

The MDeInstall Queue Manager utility run file is

[sys]<sys>MDeInstallQMgr.run

New Queue Requests

The following is a list of new BTOS operating system requests now supported by the XE 500 Queue Manager as of the 6.0 release level software:

AddQueue
ApplicationSwapped
ApplicationTerminated
DeInstallQueueManager
GetQMStatus
RemoveQueue

For information about other queue requests, refer to the *BTOS Reference Manual.*

Configuring the lpr Printer Spooler

The CENTIX lpr printer spooler is actually a BTOS-based spooler service. When the CENTIX **lpr** command is invoked to print a file, a connection is made with BTOS printer spooler queues. The queue to which the print file is sent is determined by the arguments included with the **lpr** command when it is invoked.

The default lpr printer spooler files are set up to support

- A parallel printer connected to the parallel printer port of CP00. The parallel printer name is PARALLEL and it services the BTOS print queue called "spl."
- A serial printer connected to channel 3 of CP00. The serial printer name is SERIAL and it services the BTOS print queue called "spl."

If the **lpr** command is invoked without specifying a print queue, the spl queue is used. If the **lpr** command includes the **-q** flag and a queue name is specified, the print file is sent to that printer spooler queue.

This section describes how the lpr printer spooler works and the default configurations. With this information, you can use BTOS and CENTIX utilities to modify the default configuration or create new lpr printer spooler queues.

What is a Printer Spooler?

A printer spooler is a system service that manages the transfer of data from disk files to printers.

Spooled printing allows the sharing of printers among many users. Also, while direct printing keeps the user waiting until a printer is available, spooled printing allows the user to issue a request for printing at any time and then proceed to other activities.

How the Ipr Printer Spooler Operates

The Ipr printer spooler has one or more print queues to which files are sent for printing. Each print queue is assigned one or more printers. When a printer is available, it prints out the next available file from its print queue. The files are said to be *spooled* because they are continuously sent from the print queue to the next available printer.

Because files in print queues are disk-based, the Ipr printer spooler can resume automatically after the system is rebooted.

There is no one file that can be referred to as the "Ipr printer spooler" file. The printer spooler actually comprises

- A set of printer spooler queues, which are under the control of the Queue Manager.
- An Ipr spooler configuration file.
- Printer configuration files.
- A spooler manager run statement in the initialization file of any processor to which printers are connected. This run statement causes the Ipr spooler manager to be installed in the processor at boot time.
- Line entries in the CP or TP configuration file for each RS-232 port to which a printer is connected. These line entries define operating parameters related to the port. Even though this information is overridden by the information in the printer's configuration file, the entries must still be present in this file.

lpr Printer Spooler Queues

The Queue Manager controls the following queues related to printer spooler functions:

1 A scheduling queue corresponding to each print queue.

2 A control queue for every printer that serves a print queue.

3 One spooler status queue, regardless of the number of printer spooler queues.

Sample lpr Printer Spooler Queue Entries

As an example, the default queue index file contains the following text entries to support two printers called SERIAL and PARALLEL:

```
SPL/[sys]<spl>SPL.queue/1/1
SPLB/[sys]<spl>SPLB.queue/1/1
PARALLELcontrol/[sys]<spl>parallelcontrol.queue/1/1
SERIALcontrol/[sys]<spl>serialcontrol.queue/1/1
SPOOLERSTATUS/[sys]<spl>SPOOLERSTATUS.queue/1/1
```

The **print scheduling queues** are defined in the lines

```
SPL/[sys]<spl>SPL.queue/1/1
SPLB/[sys]<spl>SPLB.queue/1/1
```

The first parameter in these entries defines the queue names to be "spl" and "splb."

The second parameter defines the default file spec names for the queue entry files. These names were arbitrarily chosen to indicate the correspondence between the name of the print scheduling queue (for example, spl) and its queue entry file.

The third parameter defines the queue sizes to be 1 sector (512 kB). As the number of queue entries grows, the size of the queue is dynamically expanded by the Queue Manager as required.

The fourth parameter defines the queues to be printer spooler related queues.

The **control queues** are defined in the lines

```
PARALLELcontrol/[sys]<spl>parallelcontrol.queue/1/1
SERIALcontrol/[sys]<spl>serialcontrol.queue/1/1
```

The first parameter in these entries define the control queue names. "PARALLELcontrol" and "SERIALcontrol" were chosen to correspond to the names of the two printers in the system, "PARALLEL" and "SERIAL." The term that precedes "control" in the control queue's name must be the printer's name as assigned in the spooler configuration file.

The second parameter defines the file spec name for the control queues. The file spec names have been arbitrarily chosen to indicate the correspondence between the name of the control queue (for example, PARALLELcontrol) and its queue entry file.

The third parameter defines the queue sizes to be 1 sector (512 kB).

The fourth parameter defines the queues to be printer spooler-related queues.

The **spooler status entry** is the line

```
SPOOLERSTATUS/[sys]<spl>SPOOLERSTATUS.queue/1/1
```

This line defines the queue name to be SPOOLERSTATUS, the file spec to be [sys]<spl>SPOOLERSTATUS.queue, the queue size to be 1 sector, and the queue to be a printer spooler-related queue.

The lpr Spooler Configuration File

There is one lpr spooler configuration file, [sys]<sys>splcnfg.sys, for all lpr printer spooler managers installed on XE 500 processors. The spooler configuration file contains an entry for each printer being used with the printer spoolers. A printer's entry defines

1 The processor channel to which the printer is connected.

2 The printer's name.

3 The print scheduling queue that serves that printer.

4 The printer's configuration file.

5 The printer's service priority level.

6 Whether a banner is to be printed before each print job sent to the printer.

7 The CP or TP board to which the printer is connected.

Each printer entry has the following format:

ChannelName/PrinterName/QueueName/PrinterConfigFile/ Priority/BannerOption/Processor

These fields are defined as follows.

ChannelName is a single character abbreviation that specifies the channel to which the printer is connected. The following table lists the CP and TP printer channels and their abbreviated names. Channels D through J apply only to a TP that is running a big printer spooler manager.

Name	Processor Channel
0	Parallel printer
A	Channel 1 (sync/async)
B	Channel 2 (sync/async)
C	Channel 3 (async only for CP; sync/async for TP)
D	Channel 4 (sync/async; TP only)
E	Channel 5 (async; TP only)
F	Channel 6 (async; TP only)
G	Channel 7 (async; TP only)
H	Channel 8 (async; TP only)
I	Channel 9 (async; TP only)
J	Channel 10 (async; TP only)

PrinterName is the arbitrary name of the printer. All lpr printers in your installation must have unique names. Also, remember that the printer name chosen here and the one used in the name for that printer's control queue in the queue index must match. For example, if you have named the printer "PRINTER1," the control queue name must be "PRINTER1control."

QueueName is the name of the printer's scheduling queue. This name must match the name of the scheduling queue listed in the queue index that you want the printer to serve. A scheduling queue can have more than one printer serving it.

PrinterConfigFile is the name of the printer's configuration file.

Priority is a numeric value that determines the priority of the printing job going to the printer in relation to other processes running on the processor. Allowable priority values range from 10 to 254, with 10 being the highest priority.

BannerOption is a yes (Y) or no (N) value that determines whether a banner is to be printed at the beginning of each printing job:

Y—Print banner at the beginning of each file.

N—No banner page is printed.

Processor is the name of the processor board to which the printer is connected (for example, CP00, TP01).

Default Spooler Configuration File

The following default spooler configuration file is configured to support two printers, called SERIAL and PARALLEL.

```
0/PARALLEL/SPL/[sys]<spl>SPLconfigCp00.sys/64/y/Cp00
C/SERIAL/SPLB/[sys]<spl>SPLCconfigCp00.sys/64/y/Cp00
```

From the first entry we know the following information about its corresponding printer:

1 The printer is connected to the parallel printer channel.

2 The printer's name is "PARALLEL." The printer's control queue name in the queue index file is, therefore, PARALLELcontrol.

3 The printer services the spl scheduling queue.

4 The printer's configuration file name is [sys]<spl>SPLconfigCp00.sys.

5 The printer's priority level is 64.

6 A banner will be printed for each printing task sent to the printer.

7 The printer is connected to CP00.

Note that any CP or TP channels assigned to printers by spooler configuration file entries are reserved exclusively for printing.

If a CP or TP channel is not specified in a spooler configuration file entry, the channel can be used by other programs.

Printer Configuration File

The printer configuration file defines the line characteristics for the channel to which the printer is connected.

Note: *The information in this file overrides the channel's line characteristics as defined in the processor's configuration file. However, the configuration file for the processor to which the printer is connected must still have a line entry for the printer.*

A printer configuration file can be created or modified by using the BTOS MCreate Configuration File utility. A description of this utility is provided in Section 20.

What the lpr Printer Spooler Does

There are three types of operations that are controlled through the lpr printer spooler service:

1 Requesting that a file be printed.

2 Controlling an active printing task (for example, cancelling the task, halting the task, resuming the task after a halt).

3 Requesting spooler status information.

When an lpr print request is made, the request is placed in the specified scheduling queue. Requests are processed in order and print files are sent out to the printer that is servicing the selected queue.

If more than one printer is servicing a queue, the print file is sent to the first available printer servicing that queue. The information contained in the spooler and printer configuration files is used to identify the current spooler configuration and the location and operating parameters of the printers.

When a control operation is invoked on a particular printer by means of the BTOS MSpooler Status utility, the control request is placed in the printer's control queue. Control requests are then processed in order.

The printer spooler places status information on each printer into the spooler status queue for use by the MSpooler Status utility.

Modifying the lpr Printer Spooler Operation

When you want to change your system's lpr printer spooler operations (for example, to add a printer, remove a printer, install a printer spooler on a CP or TP), you must make modifications to spooler-related files. These tasks are discussed below.

Note: *When you modify or create configuration files, you must reboot the system in order for the system to implement the changes.*

Adding a Printer

When adding a printer to the printer spooler configuration,

1 Use the MCreate Configuration File utility to create the printer configuration file for the printer.

2 Use an editor to add an entry for the printer in the spooler configuration file.

3 Use the ofed or ofvi editor to add an entry for the printer's control queue to the queue index file.

4 If the printer is to have its own schedule queue, use the ofed or ofvi editor to add an entry for that schedule queue to the queue index.

5 If adding a serial printer, use the MBTOS Config utility to add the appropriate async entry to the processor's configuration file. (An entry must exist in the processor configuration file even though this information is overridden by the information in the printer's configuration file.)

Note: *If more than one printer is servicing a schedule queue, a queued print file is sent to the first available printer servicing that queue.*

For this reason it is recommended that, if the format of the printout is critical, all printers servicing a particular queue be of the same type.

Moving a Printer

When moving a printer within the system (for example, to a different channel, processor, or scheduling queue),

1 Be sure that the CP or TP to which the printer is to be connected is running a printer spooler manager.

2 Modify the printer's entry in the spooler configuration file to reflect the printer's new location and/or scheduling queue.

3 If moving a serial printer, use the MBTOS Config utility to add the appropriate async entry to the configuration file of the printer's new processor. (An entry must exist in the processor configuration file even though this information is overridden by the information in the printer's configuration file.)

   Also, use the MBTOS Config utility to remove the appropriate async entry from the configuration file of the printer's old processor.

Removing a Printer

When removing a printer from the system, it is necessary only to remove that printer's entry in the spooler configuration file. It is logical but not necessary to remove the printer's scheduling queue from the queue index file if it is the only printer servicing that queue, to remove the printer's control queue from the queue index file, and to delete the printer's configuration file.

Correcting Corrupted Print Queue Files

If a file containing nonprintable characters is sent to a printer, the printer may stop when a nonprintable character is encountered. This would also stop the printing of all other files in the print queue indefinitely.

This problem is corrected by removing the corresponding queue entry file (queue entry files are defined in the queue index file). An empty queue entry file is then recreated the next time the system is rebooted. This causes the queue to be cleared.

The queue entry file can be removed by rebooting the system from a bootable disk cartridge and deleting the file.

Use the following procedure to clear the problem print queue:

1 Using the MBTOS Config utility, remove the Queue Manager entry, "$run [sys]<sys>MInstallQMgr.run" from the master processor's restricted mode initialization file.

2 Shutdown CENTIX using the **shutdown** or **halt** command.

3 Reboot the system by turning the base enclosure keyswitch to STOP and then to REMOTE.

4 Use the BTOS MDelete utility to remove the queue entry file that corresponds to the problem print queue.

5 Using the MBTOS Config utility, add the Queue Manager entry, "$run [sys]<sys>MInstallQMgr.run" back to the master processor's restricted mode initialization file.

6 Shutdown the restricted mode CENTIX system by selecting the Shutdown operation from the Restricted Mode Main Menu.

7 Reboot the system to normal mode by turning the base enclosure keyswitch to STOP and then to NORMAL.

If you are running multiple AdminAgents and have a slave AdminAgent running on the same processor as the Queue Manager, you can use the following procedure to clear the problem print queue:

1 Make sure no one on the system is using the Queue Manager.

2 Login into the system as superuser.

3 Following the shell prompt, enter

```
# ofcli -f Xpnn
```

where *Xpnn* (for example, FP00, CP01) is the processor on which the Queue Manager is running. Then press the RETURN key. The CLI prompt ($) is displayed.

4 Following the CLI prompt, enter

```
$run [sys]<admin>MDelnstallQMgr.run
```

and press the RETURN key.

The Queue Manager is removed from the specified processor.

5 Using the BTOS MDelete utility, delete the BTOS queue entry file that corresponds to the problem print queue.

6 Following the shell prompt, enter

```
# ofcli -f Xpnn
```

where *Xpnn* (for example, FP00, CP01) is the processor on which the Queue Manager was running. Then press the RETURN key. The CLI prompt ($) is displayed.

7 Following the CLI prompt, enter

```
$run [sys]<admin>MInstallQMgr.run
```

and press the RETURN key.

The Queue Manager is reinstalled on the specified processor.

lpr Printer Translation Files

The MMake Translation File utility allows you to translate characters in a document into different characters or printer control sequences as you print a document. It is intended primarily for inserting printer function control (escape) sequences in text. You can use it to translate any character into any other single character, or a single character into a series of characters.

Note: *It is possible to use the MMake Translation File utility for text substitution (for example, substituting the word "ONE" for the number "1"). However, this can interfere with the system's ability to determine when it has reached the line length specified in the printer device configuration file (refer to Sections 12 and 20). Therefore, you should not use translation files for text substitution.*

You use the ofed or ofvi editors to create a source file that lists the characters you want to translate. When the source file is complete, execute the BTOS MMake Translation File utility. This utility creates a translation file from the source file.

You can then specify the translation file when you create or modify the printer device configuration file (refer to Section 20).

Creating the Source File

To create a source file, use the ofed or ofvi editor. Make an entry for each character you want to translate, using ASCII hexadecimal (hex) codes to represent the characters. Entries should be in the following format:

XX = *YY*

where *XX* is the ASCII hex code for the character you want to translate, and *YY* is the ASCII hex code for the translated character.

For example, if you have an AP 1311 printer, to translate the pound sign (#) into the printer function control code Select 10 cpi, make the following entry:

23 = 1D

Note: *Refer to your printer's documentation for a listing of printer function control codes.*

A translation file that converts all lower case letters to upper case is included with the standard XE 500 BTOS system software. The translation file name is

[sys]<spl>LowerToUpper.sys

The source file for this translation file has also been included. It provides an example of how a translation source file is structured. The translation source file is

[sys]<spl>LowerToUpper.txl

Creating a Translation File

Use the BTOS MMake Translation File utility to convert a source file into a translation file. Refer to the *XE 500 BTOS Operations Reference Manual* for information about the run file and parameters for this utility. Refer to the *XE 500 CENTIX Administration Guide* for information about executing BTOS utilities from CENTIX.

Creating I/O Device Configuration Files

XE 500 system printers, half-inch magnetic tape drives, QIC tape drive, and data communication (data comm) equipment (for example, modems) require configuration files to define their hardware and software operating parameters. Refer to Section 12 for a description of the default I/O device configuration files.

To create or modify a device configuration file, invoke the BTOS MCreate Configuration File utility through the CENTIX **ofcli** command or centrEASE. The run file for this utility is

[sys]<admin>MCreateConfigurationFile.run

The MCreate Configuration File utility is a menu-driven utility that prompts for the configuration file name, the device type, and the parameter values necessary to configure that device into the system. When you have finished entering the parameters for a device, the utility creates the configuration file or, if modifying an existing file, overwrites the existing file with the new configuration file.

Before running the MCreate Configuration File utility, you should have the device's supporting documentation to determine the proper parameter values.

I/O Device Configuration File Names

I/O device configuration files are normally assigned names that identify the device. For example, printer configuration files can be assigned names that identify the processor and channel to which they are connected. The printer configuration file name

[sys]<spl>SPLCConfigTp00.sys

was selected for the printer connected to channel C (that is, channel 3) of processor TP00. The characters "SPLC" are included to indicate that the printer supports the lpr spooler queue splc. The term ".sys" is appended to indicate a system service file. The file is stored in the directory <spl> with other lpr spooler-related files.

Using the MCreate Configuration File Utility

The MCreate Configuration File utility is a menu-driven utility through which a configuration file for an XE 500 I/O device can be created. I/O devices include serial printers, parallel printers, tape drives, and modems.

Default Parameter Values

When creating a new configuration file, standard parameter default values are displayed in square brackets ([]) at the end of each prompt.

Note: *The default values shown in the following description of this command are not necessarily the default values provided by your software release.*

When modifying a previously created configuration file, the parameter values currently in the configuration file are displayed in the square brackets, rather than the standard default values. This allows you to change one or more parameters in the file without having to reenter values for all of the parameters.

Prompt Instructions

After being invoked, the MCreate Configuration File utility displays the following prompt:

```
Enter Configuration File Name:
```

Enter the name of the new configuration file to be created, or the name of an existing file to be modified, and press the RETURN key.

The following prompt appears:

```
Tape, QIC, Parallel Printer, Serial Printer, Communications:
```

Here you must select the device type for the device for which you are creating a configuration file. Specify the first letter for one of the above device types, that is, "t," "q," "p," "s," or "c," and press the RETURN key.

After you select a device type, the following message appears:

```
<CR> to use [Default], <GO>/ESC when all satisfied.
```

The system also displays the first parameter prompt for the configuration file that you are creating or modifying. For this and for subsequent prompts, you can enter a new value or you can accept the default value, displayed in square brackets, by not entering any value. To advance to the next prompt, press the RETURN key. If you press the RETURN key after the last prompt, you loop back to the first prompt. When you have entered the values you want for all of the prompts, press the GO key. The following line appears on the screen:

```
Continue to Configure?
```

To create or modify another configuration file, press the GO key. You will be prompted for input as before. If you do not want to create or modify another configuration file, press the FINISH key, and you will exit the MCreate Configuration File utility.

A different set of prompts is displayed for each device type selected. Below are descriptions of the different sets of prompts.

Tape Parameter Prompts

Record Size [1024]	This is the record size for fixed-length records and the maximum size for variable-length records. Only even-value record sizes can be used. The maximum value that can be used is 16384 (16 kb).
High Speed Tape [No]	Enter "yes" for the running at 100 IPS (inches per second) and "no" for 25 IPS. System configuration changes are needed to utilize high speed tape.
Rewind Tape on Close [Yes]	Enter "yes" to have the tape automatically rewind when a tape operation is completed.
Variable Length Records [No]	Enter "yes" if you want to specify different record sizes in different tape requests.

QIC Tape Parameter Prompts

Record Size [8192]	This is the maximum size of a tape record. Currently, the value 8192 is the only valid record size value for QIC tapes.
Rewind Tape on Close [No]	Enter "yes" to have the tape automatically rewind when a tape operation is completed.

Parallel Printer Parameter Prompts

TAB Expansion Size [8]	Enter the number of blank spaces into which a tab character is to be converted.
Characters Per Line [132]	This is the maximum number of characters in a print line.
Transmit Time Out - number of seconds [0]	This is the number of seconds a write operation waits to begin transmitting a character to the printer before the error code 300 (device not ready) is returned. This value should be set to any value other than 0.
Additional ACK Delay - number of 100 microseconds [0]	Specify the extra amount of time (beyond the normal 10-20 microseconds) for the system to wait between receiving an ACK signal from the printer and sending a character to the printer. This is required for some printers that do not fully conform to the XE 500 Centronics interface.
Newline Mapping Mode - Binary, CR, or CR/LF [CR/LF]	This parameter determines the mapping, or translation, of carriage return characters (hex 0A) sent from the system to the printer. Binary causes no translation to be done. CR causes a translation to hex 0D (an ASCII carriage return). CR/LF causes a translation to hex 0D,0A (an ASCII carriage return and line feed).

Current Translation File	Enter the name of the translation file that is currently being used to perform character translation. Leave this field blank if character translation is not being performed.
Change Translation File to	Enter the name of the translation file that is to be used to perform character translation. Leave this field blank if character translation is not being performed. Refer to Section 19 for a discussion of the BTOS MMake Translation File utility and how a translation file can be created.

Serial Printer Parameter Prompts

Data Bits - 5, 6, 7 or 8 [7]	This is the number of bits per character, excluding parity bits.
Baud Rate - up to 19200 [9600]	This is the transmission speed. Common values are 110, 150, 300, 1200, 2400, 4800, and 9600.
Stop Bits - 1 or 2 [1]	This is the number of stop bits per character.
Transmit Time Out - number of Seconds [0]	This is the number of seconds a write operation on the system waits to begin transmitting a character to the printer before the erc 300 "Device not ready" is returned. This value should be set to any value other than 0.
Characters per line [132]	This is the maximum number of characters in a print line.
TAB expansion size [8]	Enter the number of blank spaces into which a tab character is to be converted.
Parity - None, Even, Odd, 0 or 1 [Even]	This is the state of the parity bit.
Line Control - None, Xon/Xoff, CTS, or Both [Xon/Xoff]	This parameter determines the line protocol to be used with the serial printer. None causes no line control to be established. Xon/Xoff stops transmission when the serial printer receives an Xoff character (hex 13); when it receives an Xon character (hex 011), it resumes transmission. CTS suspends transmission when the serial printer does not receive the CTS signal from the XE 500.

Newline Mapping Mode—Binary, CR, or CR/LF [CR/LF]	This parameter determines the mapping, or translation, of carriage return characters (hex 0A) sent from the system to the printer. Binary causes no translation to be done. CR causes a translation to hex 0D (an ASCII carriage return). CR/LF causes a translation to hex 0D,0A (an ASCII carriage return and line feed).
Current Translation File	Enter the name of the translation file that is currently being used to perform character translation. Leave this field blank if character translation is not being performed.
Change Translation File to	Enter the name of the translation file that is to be used to perform character translation. Leave this field blank if character translation is not being performed. Refer to Section 19 for a discussion of the BTOS MMake Translation File utility and how a translation file can be created.

Communications Device Parameter Prompts

Data Bits—5, 6, 7 or 8 [7]	This is the number of bits per character, excluding parity characters.
Baud Rate—up to 19200 [9600]	This is the transmission speed. Common values are 110, 150, 300, 1200, 2400, 4800, and 9600.
Stop Bits—1 or 2 [1]	This is the number of stop bits per character.
Transmit Time Out—number of Seconds [0]	This is the number of seconds a write operation on the system waits to begin transmitting a character to the device before the error code 300 "Device not ready" is returned.
Receive Time Out—number of Seconds [65535]	Enter the number of seconds a read operation on the system is to wait for data before returning the error code 602 "No character available."
End of File Byte—None or Hex Value [04]	Enter the hex value which will indicate to the communications device that it has reached the end of the file that it is receiving. Entering "None" causes the communications device to never detect the end of a file which it is receiving.
Parity—None, Even, Odd, 0 or 1 [0]	This parameter determines the state of the parity bit.

Line Control—None, Xon/Xoff, CTS, or Both [Xon/Xoff]	This parameter determines the line protocol to be used by the communications device. None causes no line control to be established. Xon/Xoff stops transmission when the communications device receives an Xoff character (hex 13); when it receives an Xon character (hex 011), it resumes transmission. CTS suspends transmission when the communication device does not receive the CTS signal.
Newline Mapping Mode—Binary, CR, or CR/LF [CR/LF]	This parameter determines the mapping, or translation, of carriage return characters (hex 0A) sent from the system to the communications device. Binary causes no translation to be done. CR causes a translation to hex 0D (an ASCII carriage return). CR/LF causes a translation to hex 0D,0A (an ASCII carriage return and line feed).
CR/LF Mapping Mode—Binary or Newline [Newline]	This parameter determines the mapping, or translation, of CR/LF characters sent from the communication device to the system. Binary causes no translation to be done. Newline causes a CR, LF, or a CR/LF pair to be translated into a carriage return (hex 0A).

Maximum Tape Record Size for MTape Utilities

When using some BTOS MTape utilities, the maximum record size cannot exceed 10581 bytes. The actual maximum record size may be less depending on the amount of processor memory available to the utility at execution time.

If the tape record size exceeds the actual maximum when an MTape utility is being executed, the utility fails and the BTOS error code 9080 is reported.

If you get this error code when running a BTOS MTape utility, you can try moving system services from the processor on which the Tape Server is running to free up processor memory. Do not try to modify the buffer size specified in the Tape Server run statement. The maximum record size is dependent on how the MTape utilities work; it is not dependent on the Tape Server parameters.

Software Index

This section lists the XE 500 BTOS system files that are installed during the installation procedure.

You can check the entries in the Boot Load and MSysLoad log files against this list to see if the appropriate BTOS files were installed. If you selectively installed system files during MSysLoad, your system may not contain all of the BTOS files listed here.

This section also includes a listing of the CENTIX files included in the 8-user Run-Time System, the 16- and 32-user Run-Time System, and the Development System Extension.

Note: *The CENTIX file listings do not include files in the /SAF or /usr/lib/terminfo. These files are not executable files.*

Listing for XE 500 Boot Load BTOS Software

Restricted Mode Software

[sys]<admin>MSysLoad.run
[sys]<admin>MBtosConfig.run
[sys]<sys>Master.R.cnf
[sys]<sys>InitFp00.R.jcl
[sys]<sys>InitCp00.R.jcl
[sys]<sys>InitDp00.R.jcl
[sys]<sys>InitSp00.R.jcl
[sys]<sys>SysImage.sys
[sys]<sys>CpBtos.R.sys
[sys]<sys>DpBtos.R.sys
[sys]<sys>SpBtos.R.sys
[sys]<sys>TpBtos.R.sys
[sys]<sys>Fp00.R.cnf
[sys]<sys>Fp01.R.cnf
[sys]<sys>Fp02.R.cnf
[sys]<sys>Dp00.R.cnf
[sys]<sys>Dp01.R.cnf
[sys]<sys>Cp00.R.cnf
[sys]<sys>Sp00.R.cnf
[sys]<sys>Tp00.R.cnf
[sys]<backup>MBootCmds.sub
[sys]<sys>Cli.run
[sys]<sys>MfAdminAgent.run
[sys]<sys>MRequest.9.sys
[sys]<sys>MRequest.Q.sys
[sys]<sys>Request.Q.sys
[sys]<Admin>MIvolume.run
[sys]<sys>IvMemorex166.cnf
[sys]<sys>IvAtasi46.cnf
[sys]<sys>IvMicropolis85.cnf
[sys]<sys>IvToshiba85.cnf
[sys]<Admin>MRestore.run
[sys]<Admin>MTapeRestore.run
[sys]<Admin>MInstallServer.run
[sys]<sys>MInstallSecondary.run

[sys]<Admin>MCreatePartition.run
[sys]<Admin>MDelete.run
[sys]<Admin>MCopy.run
[sys]<Admin>MFiles.run
[sys]<sys>Queue.Index
[sys]<sys>MSpoolerMgr.run
[sys]<sys>MInstallQMgr.run
[sys]<sys>MQueueMgr.run
[sys]<sys>SplCnfg.sys
[sys]<spl>SplConfigCp00.sys
[sys]<spl>SplCConfigCp00.sys
[sys]<sys>TmpFilter.run
[sys]<sys>DEFinitcb00.jcl
[sys]<sys>DEFinitcc00.jcl
[sys]<sys>DEFinitdb00.jcl
[sys]<sys>DEFinitdc00.jcl
[sys]<sys>DEFinitfb00.jcl
[sys]<sys>DEFinitfc00.jcl
[sys]<sys>DEFinitsb00.jcl
[sys]<sys>DEFinitsc00.jcl
[sys]<sys>DEFinittb00.jcl
[sys]<sys>DEFinittc00.jcl
[sys]<sys>DEFcb00.cnf
[sys]<sys>DEFcb01.cnf
[sys]<sys>DEFcc00.cnf
[sys]<sys>DEFcc01.cnf
[sys]<sys>DEFdb00.cnf
[sys]<sys>DEFdb01.cnf
[sys]<sys>DEFdc00.cnf
[sys]<sys>DEFdc01.cnf
[sys]<sys>DEFfb00.cnf
[sys]<sys>DEFfb01.cnf
[sys]<sys>DEFfb02.cnf
[sys]<sys>DEFfb03.cnf
[sys]<sys>DEFfc00.cnf
[sys]<sys>DEFfc01.cnf
[sys]<sys>DEFfc02.cnf
[sys]<sys>DEFfc03.cnf
[sys]<sys>DEFsb00.cnf
[sys]<sys>DEFsc00.cnf
[sys]<sys>DEFtb00.cnf
[sys]<sys>DEFtc00.cnf
[sys]<sys>BootLoadParams

Master Commands

[sys]<sys>Signon.txt
[sys]<cmd>WsAdminAgent.txt
[sys]<admin>WsAdminAgent.run
[sys]<backup>XE500Cmds.sub
[sys]<backup>MSpoolerCmds.sub
[sys]<backup>TapeCmds.sub
[sys]<backup>QicCmds.sub

Tape Server

[sys]<admin>MtapeServer.run
[sys]<sys>tapeconfig.sys
[sys]<cmd>MTapeServer.rules

Miscellaneous Files—BLD

[sys]<sys>Cli.run
[sys]<sys>MCreatePartition.run
[sys]<sys>MInstallServer.run
[sys]<sys>MInstallSecondary.run
[sys]<sys>MfAdminAgent.run
[sys]<sys>MRequest.9.sys
[sys]<sys>MRequest.q.sys
[sys]<sys>Request.q.sys
[sys]<sys>MIvolume.run
[sys]<sys>IvMemorex166.cnf
[sys]<sys>IvAtasi46.cnf
[sys]<sys>IvMicropolis85.cnf
[sys]<sys>IvToshiba85.cnf
[sys]<sys>IvUnknown.cnf
[sys]<sys>MChgVolName.run
[sys]<sys>BootLoad.A.run
[sys]<sys>Master.cnf
[sys]<sys>SysImage.sys
[sys]<sys>Fp00.cnf
[sys]<sys>InitFp00.jcl

Miscellaneous Files—BLC

[sys]<sys>Cli.run
[sys]<sys>MCreatePartition.run
[sys]<sys>MInstallServer.run
[sys]<sys>MInstallSecondary.run
[sys]<sys>MfAdminAgent.run
[sys]<sys>MRequest.9.sys
[sys]<sys>MRequest.q.sys
[sys]<sys>Request.q.sys
[sys]<sys>MIvolume.run
[sys]<sys>IvMemorex166.cnf
[sys]<sys>IvAtasi46.cnf
[sys]<sys>IvMicropolis85.cnf
[sys]<sys>IvToshiba85.cnf
[sys]<sys>IvUnknown.cnf
[sys]<sys>MChgVolName.run
[sys]<sys>BootLoad.C.run
[sys]<sys>Master.cnf
[sys]<sys>SysImage.sys
[sys]<sys>Fp00.cnf
[sys]<sys>InitFp00.jcl

Miscellaneous Files—BLT

[sys]<sys>Cli.run
[sys]<sys>MCreatePartition.run
[sys]<sys>MInstallServer.run
[sys]<sys>MInstallSecondary.run
[sys]<sys>MfAdminAgent.run
[sys]<sys>MRequest.9.sys
[sys]<sys>MRequest.q.sys
[sys]<sys>Request.q.sys
[sys]<sys>MIvolume.run
[sys]<sys>IvMemorex166.cnf
[sys]<sys>IvAtasi46.cnf
[sys]<sys>IvMicropolis85.cnf
[sys]<sys>IvToshiba85.cnf

[sys]<sys>IvUnknown.cnf
[sys]<sys>MChgVolName.run
[sys]<sys>BootLoad.E.run
[sys]<sys>Master.cnf
[sys]<sys>SysImage.sys
[sys]<sys>Dp00.cnf
[sys]<sys>InitDp00.jcl

CENTIX Restricted Mode Software

[sys]<sys>ConfigUfs.R.sys
[sys]<sys>InitFp00.R.jcl
[sys]<admin>MCreateSwapR.run

centrEASE Software

[sys]<sys>Partition.R.2

PT 1500 Support

[sys]<sys>Ws100>SysImage.sys
[sys]<sys>Ws101>SysImage.sys

Listing for BTOS in XE 500 CENTIX

Queue Manager

[sys]<sys>MDeInstallQMgr.run
[sys]<sys>MInstallQMgr.run
[sys]<sys>MQueueMgr.run
[sys]<cmd>MInstallQMgr.rules

Tape Server

[sys]<admin>MtapeServer.run
[sys]<sys>tapeconfig.sys
[sys]<cmd>MTapeServer.rules

QIC Server

[sys]<admin>MQicServer.run
[sys]<sys>qicconfig.sys
[sys]<admin>MQicRetension.run
[sys]<admin>MQicErase.run
[sys]<cmd>MQicServer.rules

BTOS Tape Utilities

[sys]<admin>MTapeSelectiveBackup.run
[sys]<admin>MTapeBackupVolume.run
[sys]<admin>MTapeRestore.run
[sys]<admin>MTapeCopy.run
[sys]<admin>MTape.lib

lpr Printer Spooler

[sys]<sys>MspoolerMgr.run
[sys]<sys>MBigSpoolerMgr.run
[sys]<admin>MSpoolerStatus.run
[sys]<spl>SplConfigCp00.sys
[sys]<spl>SplCConfigCp00.sys
[sys]<sys>SplCnfg.sys
[sys]<admin>MMakeTxlFile.run
[sys]<spl>LowerToUpper.sys
[sys]<spl>LowerToUpper.txl
[sys]<cmd>MBigSpoolerMgr.rules
[sys]<cmd>MSpoolerMgr.rules
[sys]<sys>Queue.Index

BTOS Utilities

[sys]<sys>MfAdminAgent.run
[sys]<sys>SysAdminAgent.run
[sys]<sys>SlvAdminAgent.run
[sys]<sys>Request.q.sys
[sys]<sys>MRequest.q.sys
[sys]<sys>MRequest.9.sys
[sys]<admin>MVolReport.run
[sys]<admin>MDiskVerify.run
[sys]<admin>MIVolume.run
[sys]<sys>lvSyquest6.cnf
[sys]<sys>lvAtasi46.cnf
[sys]<sys>lvMemorex166.cnf
[sys]<sys>lvMicropolis85.cnf
[sys]<sys>lvToshiba85.cnf
[sys]<admin>MCreateConfigurationFile.run
[sys]<admin>MCreateDirectory.run
[sys]<admin>MRemoveDirectory.run
[sys]<admin>MSetDirectoryProtection.run
[sys]<admin>MSetFileProtection.run
[sys]<admin>MCopy.run
[sys]<admin>MDelete.run
[sys]<admin>MChgVolName.run

[sys]<admin>MSelectiveBackup.run
[sys]<admin>MBackupVolume.run
[sys]<admin>MRestore.run
[sys]<admin>MMaintainFile.run
[sys]<admin>MPlog.run
[sys]<admin>MRename.run
[sys]<admin>MVolumeStatus.run
[sys]<admin>MFiles.run
[sys]<admin>MVersion.run
[sys]<admin>MCreatePartition.run
[sys]<admin>MRemovePartition.run
[sys]<admin>MVacatePartition.run
[sys]<admin>MParStat.run
[sys]<admin>MInstallServer.run
[sys]<sys>MInstallSecondary.run
[sys]<admin>MCdtio.run
[sys]<admin>MDisableCluster.run
[sys]<admin>MResumeCluster.run
[sys]<sys>Cli.run
[sys]<sys>MHisto.run
[sys]<sys>MHistoProc.run
[sys]<cmd>MfAdminAgent.rules
[sys]<cmd>SlvAdminAgent.rules
[sys]<cmd>SysAdminAgent.rules

PT 1500 Support

[sys]<sys>Ws100>SysImage.sys
[sys]<sys>Ws101>SysImage.sys

CENTIX File System Server

[sys]<sys>UFS.run
[sys]<cmd>UFS.rules

lp Printer Spooler

[sys]<sys>LpSpooler.run
[sys]<sys>LpSpooler.config
[sys]<cmd>LpSpooler.rules

Master Processor Software

[sys]<sys>SysImage.sys

DP Software

[sys]<sys>DpBtos.sys

CP Software

[sys]<sys>CpBtos.sys

SP Software

[sys]<sys>SpBtos.sys

TP Software

[sys]<sys>TpBtos.sys

XE 500 BTOS Customizer

[sys]<OSBuild>FpBtosPrefix.asm
[sys]<OSBuild>CpBtosPrefix.asm
[sys]<OSBuild>SpBtosPrefix.asm
[sys]<OSBuild>TpBtosPrefix.asm
[sys]<OSBuild>DpBtosPrefix.asm
[sys]<OSBuild>CpCentixPrefix.asm
[sys]<OSBuild>DpCentixPrefix.asm
[sys]<OSBuild>FpCentixPrefix.asm
[sys]<OSBuild>FpBtosPrefixDeb.asm
[sys]<OSBuild>CpBtosPrefixDeb.asm
[sys]<OSBuild>SpBtosPrefixDeb.asm
[sys]<OSBuild>TpBtosPrefixDeb.asm
[sys]<OSBuild>DpBtosPrefixDeb.asm
[sys]<OSBuild>CpCentixPrefixDeb.asm
[sys]<OSBuild>DpCentixPrefixDeb.asm
[sys]<OSBuild>FpCentixPrefixDeb.asm
[sys]<OSBuild>Sysgen.asm
[sys]<OSBuild>Sysgen.mdf
[sys]<OSBuild>MRequest.O.asm
[sys]<OSBuild>MRequest.Q.asm

[sys]<OSBuild>MRequest.9.asm
[sys]<OSBuild>Request.asm
[sys]<OSBuild>Request.mdf
[sys]<OSBuild>RqLabl.asm
[sys]<OSBuild>RqLabl.mdf
[sys]<OSBuild>Mfedf>CdtEqu.edf
[sys]<OSBuild>FpBtosSysgen.obj
[sys]<OSBuild>CpBtosSysgen.obj
[sys]<OSBuild>SpBtosSysgen.obj
[sys]<OSBuild>TpBtosSysgen.obj
[sys]<OSBuild>DpBtosSysgen.obj
[sys]<OSBuild>CpCentixSysgen.obj
[sys]<OSBuild>DpCentixSysgen.obj
[sys]<OSBuild>FpCentixSysgen.obj
[sys]<OSBuild>FpBtosSysgenDeb.obj
[sys]<OSBuild>CpBtosSysgenDeb.obj
[sys]<OSBuild>SpBtosSysgenDeb.obj
[sys]<OSBuild>TpBtosSysgenDeb.obj
[sys]<OSBuild>DpBtosSysgenDeb.obj
[sys]<OSBuild>CpCentixSysgenDeb.obj
[sys]<OSBuild>DpCentixSysgenDeb.obj
[sys]<OSBuild>FpCentixSysgenDeb.obj
[sys]<OSBuild>Request.obj
[sys]<OSBuild>RqLabl.obj
[sys]<OSBuild>XeBtos.lib
[sys]<OSBuild>Ctos.lib
[sys]<OSBuild>Dbg.lib
[sys]<OSBuild>FpBtos.fls
[sys]<OSBuild>CpBtos.fls
[sys]<OSBuild>SpBtos.fls
[sys]<OSBuild>TpBtos.fls
[sys]<OSBuild>DpBtos.fls
[sys]<OSBuild>CpCentix.fls
[sys]<OSBuild>DpCentix.fls
[sys]<OSBuild>FpCentix.fls
[sys]<OSBuild>FpBtosDeb.fls
[sys]<OSBuild>CpBtosDeb.fls
[sys]<OSBuild>SpBtosDeb.fls
[sys]<OSBuild>TpBtosDeb.fls
[sys]<OSBuild>DpBtosDeb.fls
[sys]<OSBuild>CpCentixDeb.fls
[sys]<OSBuild>DpCentixDeb.fls
[sys]<OSBuild>FpCentixDeb.fls

[sys]<OSBuild>FpBtos.jcl
[sys]<OSBuild>CpBtos.jcl
[sys]<OSBuild>SpBtos.jcl
[sys]<OSBuild>TpBtos.jcl
[sys]<OSBuild>DpBtos.jcl
[sys]<OSBuild>CpCentix.jcl
[sys]<OSBuild>DpCentix.jcl
[sys]<OSBuild>FpCentix.jcl
[sys]<OSBuild>FpBtosDeb.jcl
[sys]<OSBuild>CpBtosDeb.jcl
[sys]<OSBuild>SpBtosDeb.jcl
[sys]<OSBuild>TpBtosDeb.jcl
[sys]<OSBuild>DpBtosDeb.jcl
[sys]<OSBuild>CpCentixDeb.jcl
[sys]<OSBuild>DpCentixDeb.jcl
[sys]<OSBuild>FpCentixDeb.jcl
[sys]<OSBuild>FpBtos.sub
[sys]<OSBuild>CpBtos.sub
[sys]<OSBuild>SpBtos.sub
[sys]<OSBuild>TpBtos.sub
[sys]<OSBuild>DpBtos.sub
[sys]<OSBuild>CpCentix.sub
[sys]<OSBuild>DpCentix.sub
[sys]<OSBuild>FpCentix.sub
[sys]<OSBuild>FpBtosDeb.sub
[sys]<OSBuild>CpBtosDeb.sub
[sys]<OSBuild>SpBtosDeb.sub
[sys]<OSBuild>TpBtosDeb.sub
[sys]<OSBuild>DpBtosDeb.sub
[sys]<OSBuild>CpCentixDeb.sub
[sys]<OSBuild>DpCentixDeb.sub
[sys]<OSBuild>FpCentixDeb.sub
[sys]<OSBuild>Samgen.asm
[sys]<OSBuild>Samgen.mdf
[sys]<sys>Assembler.runfiles
[sys]<sys>Librarian.run
[sys]<sys>Linker.run

CENTIX Run-Time System Listing

/bigunix
/bin/acctcom
/bin/adb
/bin/apnum
/bin/as
/bin/basename
/bin/cat
/bin/cc
/bin/chgrp
/bin/chmod
/bin/chown
/bin/cmp
/bin/convert
/bin/cp
/bin/cpio
/bin/date
/bin/dd
/bin/df
/bin/diff
/bin/dirname
/bin/du
/bin/dump
/bin/echo
/bin/ed
/bin/env
/bin/expr
/bin/false
/bin/file
/bin/find
/bin/grep
/bin/hpio
/bin/iAPX286
/bin/id
/bin/kill
/bin/ld
/bin/line
/bin/ln
/bin/login
/bin/ls

/bin/mail
/bin/mc68k
/bin/mkdir
/bin/mv
/bin/newgrp
/bin/nice
/bin/nohup
/bin/od
/bin/passwd
/bin/pdp11
/bin/pr
/bin/ps
/bin/pwd
/bin/red
/bin/rm
/bin/rmail
/bin/rmdir
/bin/rsh
/bin/sed
/bin/sh
/bin/size
/bin/sleep
/bin/sort
/bin/stty
/bin/su
/bin/sync
/bin/tail
/bin/tee
/bin/telinit
/bin/touch
/bin/tput
/bin/true
/bin/tty
/bin/u370
/bin/u3b
/bin/u3b2
/bin/u3b5
/bin/uname

/bin/vax
/bin/wc
/bin/who
/dev/MKDEV
/etc/.findpt
/etc/.map
/etc/.mkconrc
/etc/.restore
/etc/.restoremsg
/etc/allrc
/etc/bcheckmsg
/etc/bcheckrc
/etc/bcopy
/etc/brc
/etc/brcmsg
/etc/CentixLoad
/etc/checkall
/etc/checklist
/etc/clri
/etc/config
/etc/console
/etc/crash
/etc/cron
/etc/crup
/etc/datemsg
/etc/daterc
/etc/dcopy
/etc/devnm
/etc/dfsck
/etc/ff
/etc/filesave
/etc/finc
/etc/fpsar
/etc/frec
/etc/fsck
/etc/fsdb
/etc/fuser
/etc/getty
/etc/gettydefs8
/etc/group
/etc/grpck
/etc/halt
/etc/haltmsg

/etc/icode
/etc/init
/etc/inittab00
/etc/inittab01
/etc/ioctl.syscon
/etc/ipcstructs
/etc/issue
/etc/issuept
/etc/keystate
/etc/killall
/etc/labelit
/etc/link
/etc/log/filesave.log
/etc/magic
/etc/master
/etc/mkboot
/etc/mkconrc
/etc/mkfs
/etc/mklost+found
/etc/mknod
/etc/mksym
/etc/motd8
/etc/mount
/etc/mvdir
/etc/ncheck
/etc/ofcli
/etc/ofcopy
/etc/ofdf
/etc/ofed
/etc/ofvi
/etc/passwd
/etc/pbuf
/etc/pfilemsg
/etc/profile
/etc/pwck
/etc/rc
/etc/rc.mounts
/etc/rcmsg
/etc/renice

/etc/sadp
/etc/savefile
/etc/setmnt
/etc/setuname
/etc/shutdown
/etc/shutmsg
/etc/spawnsrv
/etc/sysdef
/etc/tapesave
/etc/termcap
/etc/tsioctl
/etc/umount
/etc/unlink
/etc/update
/etc/utmp
/etc/utmp00
/etc/utmp01
/etc/utmp02
/etc/utmp03
/etc/volcopy
/etc/wall
/etc/whodo
/etc/wtmp
/etc/wtmp00
/etc/wtmp01
/etc/wtmp02
/etc/wtmp03
/etc/wtmpclean
/lib/adb.881
/lib/ccom
/lib/ccom20
/lib/ccom20.81
/lib/cpp
/lib/ifile.0407
/lib/ifile.0410
/lib/ifile.0413
/lib/ifile.0413-F
/lib/optim
/smunix
/unix20 (16- and 32-user Run-Time System only)

/usr/bin/asa
/usr/bin/at
/usr/bin/awk
/usr/bin/banner
/usr/bin/batch
/usr/bin/bdiff
/usr/bin/cancel
/usr/bin/clear
/usr/bin/comm
/usr/bin/cpset
/usr/bin/crontab
/usr/bin/ct
/usr/bin/cu
/usr/bin/cut
/usr/bin/diff3
/usr/bin/dircmp
/usr/bin/disable
/usr/bin/edit
/usr/bin/egrep
/usr/bin/enable
/usr/bin/ex
/usr/bin/fgrep
/usr/bin/getopt
/usr/bin/id
/usr/bin/ipcrm
/usr/bin/ipcs
/usr/bin/logname
/usr/bin/lp
/usr/bin/lpr
/usr/bin/lpstat
/usr/bin/more
/usr/bin/news
/usr/bin/page
/usr/bin/path
/usr/bin/print
/usr/bin/sar
/usr/bin/sdiff
/usr/bin/spawn
/usr/bin/tabs
/usr/bin/tar
/usr/bin/tr

/usr/bin/uniq
/usr/bin/uucp
/usr/bin/uulog
/usr/bin/uuname
/usr/bin/uupick
/usr/bin/uustat
/usr/bin/uuto
/usr/bin/uux
/usr/bin/vi
/usr/bin/view
/usr/bin/xargs
/usr/include/sys/acct.h
/usr/include/sys/anode.h
/usr/include/sys/asm.h
/usr/include/sys/buf.h
/usr/include/sys/callo.h
/usr/include/sys/cmap.h
/usr/include/sys/conf.h
/usr/include/sys/crtctl.h
/usr/include/sys/devcon.h
/usr/include/sys/dir.h
/usr/include/sys/dmap.h
/usr/include/sys/erc.h
/usr/include/sys/errno.h
/usr/include/sys/exch.h
/usr/include/sys/fblk.h
/usr/include/sys/file.h
/usr/include/sys/filsys.h
/usr/include/sys/fp.h
/usr/include/sys/hardware.h
/usr/include/sys/I8251.h
/usr/include/sys/icc.h
/usr/include/sys/in.h
/usr/include/sys/init.h
/usr/include/sys/ino.h
/usr/include/sys/inode.h
/usr/include/sys/iobuf.h
/usr/include/sys/ioctl.h
/usr/include/sys/ipc.h
/usr/include/sys/kdb.h
/usr/include/sys/kprintf.h

/usr/include/sys/lock.h
/usr/include/sys/lp.h
/usr/include/sys/lprio.h
/usr/include/sys/map.h
/usr/include/sys/mbuf.h
/usr/include/sys/mount.h
/usr/include/sys/msg.h
/usr/include/sys/notify.h
/usr/include/sys/opt.h
/usr/include/sys/param.h
/usr/include/sys/pilf.h
/usr/include/sys/proc.h
/usr/include/sys/pte.h
/usr/include/sys/reg.h
/usr/include/sys/rqcode.h
/usr/include/sys/rslp.h
/usr/include/sys/sem.h
/usr/include/sys/shm.h
/usr/include/sys/signal.h
/usr/include/sys/sky.h
/usr/include/sys/socket.h
/usr/include/sys/space.h
/usr/include/sys/stat.h
/usr/include/sys/stermio.h
/usr/include/sys/sysinfo.h
/usr/include/sys/syslocal.h
/usr/include/sys/sysmacros.h
/usr/include/sys/systm.h
/usr/include/sys/tape.h
/usr/include/sys/target.h
/usr/include/sys/termio.h
/usr/include/sys/text.h
/usr/include/sys/times.h
/usr/include/sys/tp.h
/usr/include/sys/trap.h
/usr/include/sys/ttold.h
/usr/include/sys/tty.h
/usr/include/sys/types.h
/usr/include/sys/ufs.h
/usr/include/sys/user.h
/usr/include/sys/ustat.h
/usr/include/sys/utsname.h

/usr/include/sys/var.h
/usr/include/sys/vlimit.h
/usr/include/sys/vm.h
/usr/include/sys/vmmac.h
/usr/include/sys/vmparam.h
/usr/include/sys/vmsysinfo.h
/usr/include/sys/vmsystm.h
/usr/include/sys/vt.h
/usr/include/sys/window.h
/usr/lib/accept
/usr/lib/acct/acctcms
/usr/lib/acct/acctcon1
/usr/lib/acct/acctcon2
/usr/lib/acct/acctdisk
/usr/lib/acct/acctdusg
/usr/lib/acct/acctmerg
/usr/lib/acct/accton
/usr/lib/acct/acctprc1
/usr/lib/acct/acctprc2
/usr/lib/acct/acctwtmp
/usr/lib/acct/chargefee
/usr/lib/acct/ckpacct
/usr/lib/acct/diskusg
/usr/lib/acct/dodisk
/usr/lib/acct/fwtmp
/usr/lib/acct/lastlogin
/usr/lib/acct/monacct
/usr/lib/acct/nulladm
/usr/lib/acct/prctmp
/usr/lib/acct/prdaily
/usr/lib/acct/prtacct
/usr/lib/acct/remove
/usr/lib/acct/runacct
/usr/lib/acct/shutacct
/usr/lib/acct/startup
/usr/lib/acct/turnacct
/usr/lib/acct/wtmpfix
/usr/lib/cron/.proto
/usr/lib/cron/at.allow
/usr/lib/cron/cron.allow
/usr/lib/cron/queuedefs

/usr/lib/diff3prog
/usr/lib/diffh
/usr/lib/expreserve
/usr/lib/exrecover
/usr/lib/exstrings
/usr/lib/lpadmin
/usr/lib/lpenq
/usr/lib/lpfx
/usr/lib/lpmove
/usr/lib/lpsched
/usr/lib/lpshut
/usr/lib/more.help
/usr/lib/mv_dir
/usr/lib/reject
/usr/lib/sa/sa1
/usr/lib/sa/sa2
/usr/lib/sa/sadc
/usr/lib/uucp/L_devices
/usr/lib/uucp/L_dialcodes
/usr/lib/uucp/L.cmds
/usr/lib/uucp/L.sys
/usr/lib/uucp/L_stat
/usr/lib/uucp/L_sub
/usr/lib/uucp/modemcap
/usr/lib/uucp/R_stat
/usr/lib/uucp/R_sub
/usr/lib/uucp/USERFILE
/usr/lib/uucp/uucico
/usr/lib/uucp/uuclean
/usr/lib/uucp/uudemon.day
/usr/lib/uucp/uudemon.hr
/usr/lib/uucp/uudemon.wk
/usr/lib/uucp/uusub
/usr/lib/uucp/uuxqt
/usr/local/bin/download
/usr/local/bin/fsize
/usr/local/bin/hd
/usr/local/bin/head
/usr/local/bin/lpset
/usr/local/bin/mkdirs
/usr/local/bin/mvtpy

/usr/local/bin/ofls
/usr/local/bin/ofrm
/usr/local/bin/perc
/usr/local/bin/pmon
/usr/local/bin/pstat
/usr/local/bin/ptclr
/usr/local/bin/qinstall
/usr/local/bin/qlist
/usr/local/bin/script
/usr/local/bin/tdl
/usr/local/bin/tset
/usr/local/bin/wm
/usr/spool/cron/crontabs/adm
/usr/spool/cron/crontabs/root
/usr/spool/cron/crontabs/sys
/usr/spool/lp/model/1305
/usr/spool/lp/model/9246
/usr/spool/lp/model/dumb
/usr/spool/lp/pstatus
/usr/spool/lp/qstatus
/usr/spool/lp/tr9246
/usr/sys/cf/built.o
/usr/sys/cf/custscript
/usr/sys/cf/custversion.o
/usr/sys/cf/dfile.normal
/usr/sys/cf/ifile.kernel
/usr/sys/cf/linesw.o
/usr/sys/cf/low.o
/usr/sys/cf/name.o
/usr/sys/oslib/lib0
/usr/sys/oslib/lib1-8
/usr/sys/oslib/lib1a
/usr/sys/oslib/lib2
/usr/sys/oslib/lib3
/usr/sys/oslib/lib4
/usr/sys/oslib/locore.o
/usr/sys/up/cleanutl.o
/usr/sys/up/forkutl.o
/usr/sys/up/mutl.o
/usr/sys/up/pgutl.o
/usr/sys/up/swp1utl.o
/usr/sys/up/swp2utl.o
/usr/sys/up/tmp1utl.o
/usr/sys/up/tmp2utl.o
/usr/sys/up/upage.o

The following files apply to the 16- and 32-user Run-Time System only:

/usr/sys20/cf/built.o
/usr/sys20/cf/custscript
/usr/sys20/cf/custversion.o
/usr/sys20/cf/dfile.normal
/usr/sys20/cf/ifile.kernel
/usr/sys20/cf/linesw.o
/usr/sys20/cf/low.o
/usr/sys20/cf/name.o
/usr/sys20/oslib/lib0
/usr/sys20/oslib/lib1
/usr/sys20/oslib/lib1a
/usr/sys20/oslib/lib2
/usr/sys20/oslib/lib3
/usr/sys20/oslib/lib4
/usr/sys20/oslib/locore.o
/usr/sys20/up/cleanutl.o
/usr/sys20/up/forkutl.o
/usr/sys20/up/mutl.o
/usr/sys20/up/pgutl.o
/usr/sys20/up/swp1utl.o
/usr/sys20/up/swp2utl.o
/usr/sys20/up/tmp1utl.o
/usr/sys20/up/tmp2utl.o
/usr/sys20/up/upage.o

CENTIX Development System Extension Listing

/bin/ar
/bin/bcheck
/bin/bs
/bin/expand
/bin/lorder
/bin/make
/bin/mesg
/bin/nm
/bin/strip
/bin/sum
/bin/tic
/bin/time
/bin/write
/cross/1sw/lib/crt0.o
/cross/1sw/lib/crt60.o
/cross/1sw/lib/libc.a
/cross/1sw/lib/libm.a
/cross/1sw/lib/libp/libc.a
/cross/1sw/lib/libp/libmalloc.a
/cross/1sw/lib/libPW.a
/cross/1sw/lib/mcrt0.o
/cross/1sw/lib/mcrt60.o
/cross/1sw/usr/lib/lib300.a
/cross/1sw/usr/lib/lib300s.a
/cross/1sw/usr/lib/lib4014.a
/cross/1sw/usr/lib/lib450.a
/cross/1sw/usr/lib/libctos.a
/cross/1sw/usr/lib/libcurses.a
/cross/1sw/usr/lib/libdev.a
/cross/1sw/usr/lib/libg.a
/cross/1sw/usr/lib/libl.a
/cross/1sw/usr/lib/libld.a
/cross/1sw/usr/lib/libmalloc.a
/cross/1sw/usr/lib/libocurse.a
/cross/1sw/usr/lib/liboldipc.a
/cross/1sw/usr/lib/libplot.a
/cross/1sw/usr/lib/libtgt.a
/cross/1sw/usr/lib/libtrans.a

/cross/1sw/usr/lib/libvt0.a
/cross/1sw/usr/lib/liby.a
/cross/1sw/usr/lib/llib-lmalloc.l
/cross/2fp/lib/crt0.o
/cross/2fp/lib/crt60.o
/cross/2fp/lib/libc.a
/cross/2fp/lib/libm.a
/cross/2fp/lib/libp/libc.a
/cross/2fp/lib/mcrt0.o
/cross/2fp/lib/mcrt60.o
/cross/2fp/usr/lib/lib300.a
/cross/2fp/usr/lib/lib300s.a
/cross/2fp/usr/lib/lib4014.a
/cross/2fp/usr/lib/lib450.a
/cross/2fp/usr/lib/libplot.a
/cross/2fp/usr/lib/libtgt.a
/cross/2fp/usr/lib/libvt0.a
/cross/2sw/lib/crt0.o
/cross/2sw/lib/crt60.o
/cross/2sw/lib/libc.a
/cross/2sw/lib/libm.a
/cross/2sw/lib/libp/libc.a
/cross/2sw/lib/libp/libmalloc.a
/cross/2sw/lib/libPW.a
/cross/2sw/lib/mcrt0.o
/cross/2sw/lib/mcrt60.o
/cross/2sw/usr/lib/lib300.a
/cross/2sw/usr/lib/lib300s.a
/cross/2sw/usr/lib/lib4014.a
/cross/2sw/usr/lib/lib450.a
/cross/2sw/usr/lib/libctos.a
/cross/2sw/usr/lib/libcurses.a
/cross/2sw/usr/lib/libl.a
/cross/2sw/usr/lib/libmalloc.a
/cross/2sw/usr/lib/libocurse.a
/cross/2sw/usr/lib/libplot.a
/cross/2sw/usr/lib/libtgt.a
/cross/2sw/usr/lib/libvt0.a
/cross/2sw/usr/lib/llib-lmalloc.l

/etc/chroot
/etc/hinv
/etc/install
/etc/prfdc
/etc/prfld
/etc/prfpr
/etc/prfsnap
/etc/prfstat
/lib/crt0.o
/lib/crt60.o
/lib/libc.a
/lib/libm.a
/lib/libp/libc.a
/lib/libp/libmalloc.a
/lib/libPW.a
/lib/mcrt0.o
/lib/mcrt60.o
/lib/sdb.881
/usr/bin/300
/usr/bin/300s
/usr/bin/4014
/usr/bin/450
/usr/bin/admin
/usr/bin/bc
/usr/bin/bfs
/usr/bin/cal
/usr/bin/calendar
/usr/bin/cb
/usr/bin/cdc
/usr/bin/cflow
/usr/bin/checkcw
/usr/bin/checkeq
/usr/bin/checkmm
/usr/bin/col
/usr/bin/comb
/usr/bin/csplit
/usr/bin/ctc
/usr/bin/ctcr
/usr/bin/ctrace
/usr/bin/cw
/usr/bin/cxref

/usr/bin/dc
/usr/bin/delta
/usr/bin/deroff
/usr/bin/diffmk
/usr/bin/efl
/usr/bin/eqn
/usr/bin/factor
/usr/bin/fsplit
/usr/bin/get
/usr/bin/greek
/usr/bin/help
/usr/bin/hp
/usr/bin/hyphen
/usr/bin/join
/usr/bin/lex
/usr/bin/lint
/usr/bin/m4
/usr/bin/man
/usr/bin/mm
/usr/bin/mmt
/usr/bin/mvt
/usr/bin/neqn
/usr/bin/newform
/usr/bin/nl
/usr/bin/nroff
/usr/bin/osdd
/usr/bin/pack
/usr/bin/paste
/usr/bin/pcat
/usr/bin/pg
/usr/bin/prof
/usr/bin/prs
/usr/bin/ptx
/usr/bin/ratfor
/usr/bin/regcmp
/usr/bin/rmchg
/usr/bin/rmdel

/usr/bin/sact
/usr/bin/sag
/usr/bin/sccsdiff
/usr/bin/sdb
/usr/bin/sno
/usr/bin/spell
/usr/bin/split
/usr/bin/tbl
/usr/bin/tc
/usr/bin/timex
/usr/bin/troff
/usr/bin/tsort
/usr/bin/ul
/usr/bin/unget
/usr/bin/units
/usr/bin/unpack
/usr/bin/val
/usr/bin/vc
/usr/bin/vedit
/usr/bin/what
/usr/bin/yacc
/usr/include/a.out.h
/usr/include/alarm.h
/usr/include/aouthdr.h
/usr/include/ar.h
/usr/include/assert.h
/usr/include/core.h
/usr/include/ctype.h
/usr/include/curses.h
/usr/include/dial.h
/usr/include/dumprestor.h
/usr/include/errno.h
/usr/include/exch.h
/usr/include/execargs.h
/usr/include/fatal.h
/usr/include/fcntl.h
/usr/include/filehdr.h
/usr/include/ftw.h
/usr/include/gdioctl.h
/usr/include/grp.h

/usr/include/ldfcn.h
/usr/include/linenum.h
/usr/include/macros.h
/usr/include/malloc.h
/usr/include/math.h
/usr/include/memory.h
/usr/include/mnttab.h
/usr/include/mon.h
/usr/include/nan.h
/usr/include/nlist.h
/usr/include/oa/sysforms.h
/usr/include/oa/wm.h
/usr/include/ocurse.h
/usr/include/prof.h
/usr/include/pwd.h
/usr/include/regexp.h
/usr/include/reloc.h
/usr/include/rje.h
/usr/include/scnhdr.h
/usr/include/search.h
/usr/include/setjmp.h
/usr/include/sgtty.h
/usr/include/signal.h
/usr/include/spawn.h
/usr/include/stand.h
/usr/include/stdio.h
/usr/include/storclass.h
/usr/include/string.h
/usr/include/symbol.h
/usr/include/syms.h
/usr/include/sys/aouthdr.h
/usr/include/sys/filehdr.h
/usr/include/sys/flock.h
/usr/include/sys/scnhdr.h
/usr/include/sys/tsioctl.h
/usr/include/syslocal.h
/usr/include/term.h
/usr/include/termio.h
/usr/include/time.h
/usr/include/tp_defs.h
/usr/include/unctrl.h
/usr/include/unistd.h
/usr/include/ustat.h
/usr/include/utmp.h

/usr/include/values.h
/usr/include/varargs.h
/usr/include/x25lib.h
/usr/lib/calprog
/usr/lib/ctrace/runtime.c
/usr/lib/dag
/usr/lib/flip
/usr/lib/fonts/ftB
/usr/lib/fonts/ftBC
/usr/lib/fonts/ftC
/usr/lib/fonts/ftCE
/usr/lib/fonts/ftCI
/usr/lib/fonts/ftCK
/usr/lib/fonts/ftCS
/usr/lib/fonts/ftCW
/usr/lib/fonts/ftFD
/usr/lib/fonts/ftG
/usr/lib/fonts/ftGI
/usr/lib/fonts/ftGM
/usr/lib/fonts/ftGR
/usr/lib/fonts/ftH
/usr/lib/fonts/ftHI
/usr/lib/fonts/ftHM
/usr/lib/fonts/ftI
/usr/lib/fonts/ftL
/usr/lib/fonts/ftLI
/usr/lib/fonts/ftPA
/usr/lib/fonts/ftPB
/usr/lib/fonts/ftPI
/usr/lib/fonts/ftR
/usr/lib/fonts/ftS
/usr/lib/fonts/ftSB
/usr/lib/fonts/ftSI
/usr/lib/fonts/ftSM
/usr/lib/fonts/ftUD
/usr/lib/help/ad
/usr/lib/help/bd
/usr/lib/help/cb
/usr/lib/help/cm
/usr/lib/help/cmds
/usr/lib/help/co

/usr/lib/help/de
/usr/lib/help/default
/usr/lib/help/ge
/usr/lib/help/he
/usr/lib/help/lib/help2
/usr/lib/help/prs
/usr/lib/help/rc
/usr/lib/help/term
/usr/lib/help/text
/usr/lib/help/un
/usr/lib/help/ut
/usr/lib/help/vc
/usr/lib/lex/ncform
/usr/lib/lex/nrform
/usr/lib/lib.b
/usr/lib/lib300.a
/usr/lib/lib300s.a
/usr/lib/lib4014.a
/usr/lib/lib450.a
/usr/lib/libctos.a
/usr/lib/libcurses.a
/usr/lib/libdev.a
/usr/lib/libg.a
/usr/lib/libl.a
/usr/lib/libld.a
/usr/lib/libmalloc.a
/usr/lib/libocurse.a
/usr/lib/liboldipc.a
/usr/lib/libplot.a
/usr/lib/libsfoa.a
/usr/lib/libtermcap.a
/usr/lib/libtermlib.a
/usr/lib/libtgt.a
/usr/lib/libtrans.a
/usr/lib/libvt0.a
/usr/lib/libwm.a
/usr/lib/liby.a
/usr/lib/lint1
/usr/lib/lint2

/usr/lib/llib-lc
/usr/lib/llib-lc.ln
/usr/lib/llib-lm
/usr/lib/llib-lm.ln
/usr/lib/llib-lmalloc.l
/usr/lib/llib-port
/usr/lib/llib-port.ln
/usr/lib/macros/an
/usr/lib/macros/cmp.n.d.an
/usr/lib/macros/cmp.n.d.m
/usr/lib/macros/cmp.n.t.an
/usr/lib/macros/cmp.n.t.m
/usr/lib/macros/cmp.t.d.an
/usr/lib/macros/cmp.t.d.m
/usr/lib/macros/cmp.t.t.an
/usr/lib/macros/cmp.t.t.m
/usr/lib/macros/mmn
/usr/lib/macros/mmt
/usr/lib/macros/osdd
/usr/lib/macros/ptx
/usr/lib/macros/ucmp.n.an
/usr/lib/macros/ucmp.n.m
/usr/lib/macros/ucmp.t.an
/usr/lib/macros/ucmp.t.m
/usr/lib/macros/vmca
/usr/lib/manprog
/usr/lib/nmf
/usr/lib/spell/compress
/usr/lib/spell/hashcheck
/usr/lib/spell/hashmake
/usr/lib/spell/hlista
/usr/lib/spell/hlistb
/usr/lib/spell/hstop
/usr/lib/spell/spellin
/usr/lib/spell/spellprog
/usr/lib/suftab
/usr/lib/tabset/3101
/usr/lib/tabset/beehive
/usr/lib/tabset/diablo
/usr/lib/tabset/std
/usr/lib/tabset/teleray
/usr/lib/tabset/vt100
/usr/lib/tabset/xerox1720

/usr/lib/term/tab2631
/usr/lib/term/tab2631-c
/usr/lib/term/tab2631-e
/usr/lib/term/tab300
/usr/lib/term/tab300-12
/usr/lib/term/tab300s
/usr/lib/term/tab300s-12
/usr/lib/term/tab37
/usr/lib/term/tab382
/usr/lib/term/tab4000a
/usr/lib/term/tab450
/usr/lib/term/tab450-12
/usr/lib/term/tab832
/usr/lib/term/taba1
/usr/lib/term/tablp
/usr/lib/term/tabpt
/usr/lib/term/tabtn300
/usr/lib/term/tabX
/usr/lib/tmac/tmac.an
/usr/lib/tmac/tmac.m
/usr/lib/tmac/tmac.org
/usr/lib/tmac/tmac.osd
/usr/lib/tmac/tmac.ptx
/usr/lib/tmac/tmac.v
/usr/lib/unittab
/usr/lib/xcpp
/usr/lib/xpass
/usr/lib/yaccpar
/usr/local/bin/csh
/usr/local/bin/ctags
/usr/local/bin/fold
/usr/local/bin/includes
/usr/local/bin/mkstr
/usr/local/bin/petal
/usr/local/bin/strings
/usr/local/bin/xstr
/usr/pub/eqnchar

Appendix B

Default System Configuration Files

MBTOS Config General Default Initialization Files

[sys]<sys>DefInitFc00.jcl (Master FP Systems with Disk Cartridge)

```
$continueonerror
$run [sys]<sys>MInstallQMgr.run
$run [sys]<admin>MCreatePartition.run,150k,UFS
$run [sys]<admin>MInstallServer.run,UFS,[sys]<sys>UFS.run
$run [sys]<sys>MfAdminAgent.run
```

[sys]<sys>DefInitFc00.jcl (Master FP Systems with QIC Tape)

```
$continueonerror
$run [sys]<sys>MInstallQMgr.run
$run [sys]<admin>MCreatePartition.run,150k,UFS
$run [sys]<admin>MInstallServer.run,UFS,[sys]<sys>UFS.run
$run [sys]<admin>MCreatePartition.run,60k,Quick
$run [sys]<admin>MInstallServer.run,Quick,
[sys]<admin>MQicServer.run
$run [sys]<sys>MfAdminAgent.run
```

[sys]<sys>DefInitCc00.jcl

```
$run [sys]<sys>MSpoolerMgr.run
```

[sys]<sys>DefInitDc00.jcl (Master DP Systems)

```
$continueonerror
$run [sys]<sys>MInstallQMgr.run
$run [sys]<admin>MCreatePartition.run,80k,Tape
$run [sys]<admin>MInstallServer.run,Tape,
[sys]<admin>MTapeServer.run,2,16
$run [sys]<admin>MCreatePartition.run,150k,UFS
$run [sys]<admin>MInstallServer.run,UFS,[sys]<sys>UFS.run
$run [sys]<sys>MfAdminAgent.run
```

[sys]<sys>DefInitDc00.jcl (Master FP Systems)

```
$run [sys]<admin>MTapeServer.run,2,16
```

[sys]<sys>DefInitSc00.jcl

```
$run [sys]<admin>MTapeServer.run,2,16
```

[sys]<sys>DefInitTc00.jcl

```
[This file is empty.]
```

MBTOS Config General Default Configuration Files

[sys]<sys>DefFc00.cnf (Master FP Systems)

```
BlockType=y, Number=18
BlockType=z, Number=40
DISK=0  DEVICE=d0,  PASSWORD=d0
DISK=1  DEVICE=d1,  PASSWORD=d1
DISK=2  DEVICE=d2,  PASSWORD=d2
DISK=3  DEVICE=d3,  PASSWORD=d3
```

[sys]<sys>DefFc01.cnf (Master FP Systems)

```
BlockType=y, Number=18
BlockType=z, Number=40
DISK=0  DEVICE=d4,  PASSWORD=d4
DISK=1  DEVICE=d5,  PASSWORD=d5
DISK=2  DEVICE=d6,  PASSWORD=d6
DISK=3  DEVICE=d7,  PASSWORD=d7
```

[sys]<sys>DefFc02.cnf (Master FP Systems)

```
BlockType=y, Number=18
BlockType=z, Number=40
DISK=0  DEVICE=d8,  PASSWORD=d8
DISK=1  DEVICE=d9,  PASSWORD=d9
DISK=2  DEVICE=d10, PASSWORD=d10
DISK=3  DEVICE=d11, PASSWORD=d11
```

[sys]<sys>DefFc03.cnf (Master FP Systems)

```
BlockType=y, Number=18
BlockType=z, Number=40
DISK=0  DEVICE=d12, PASSWORD=d12
DISK=1  DEVICE=d13, PASSWORD=d13
DISK=2  DEVICE=d14, PASSWORD=d14
DISK=3  DEVICE=d15, PASSWORD=d15
```

[sys]<sys>DefDc00.cnf

```
BlockType=y, Number=18
BlockType=z, Number=40
DISK 0  DEVICE=s0, PASSWORD=s0
DISK 1  DEVICE=s1, PASSWORD=s1
DISK 2  DEVICE=s2, PASSWORD=s2
DISK 3  DEVICE=s3, PASSWORD=s3
DISK 4  DEVICE=s4, PASSWORD=s4
DISK 5  DEVICE=s5, PASSWORD=s5
```

[sys]<sys>DefDc01.cnf

```
BlockType=y, Number=18
BlockType=z, Number=40
DISK 0  DEVICE=s6, PASSWORD=s6
DISK 1  DEVICE=s7, PASSWORD=s7
DISK 2  DEVICE=s8, PASSWORD=s8
DISK 3  DEVICE=s9, PASSWORD=s9
DISK 4  DEVICE=s10, PASSWORD=s10
DISK 5  DEVICE=s11, PASSWORD=s11
```

[sys]<sys>DefCc00.cnf

```
BlockType=x, Number=24
BlockType=y, Number=5
BlockType=z, Number=10
async  3, speed=9600, parity=none, stopbits=1, charbits=8
line1, clock=307k
line2, clock=307k
```

[sys]<sys>DefSc00.cnf

```
BlockType=y, Number=4
BlockType=z, Number=30
```

[sys]<sys>DefTc00.cnf

```
BlockType=y, Number=4
BlockType=z, Number=30
async  5, speed=9600, parity=none, stopbits=1, charbits=8
async  6, speed=9600, parity=none, stopbits=1, charbits=8
async  7, speed=9600, parity=none, stopbits=1, charbits=8
async  8, speed=9600, parity=none, stopbits=1, charbits=8
async  9, speed=9600, parity=none, stopbits=1, charbits=8
async 10, speed=9600, parity=none, stopbits=1, charbits=8
```

Restricted Mode System Configuration Files

[sys]<sys>master.r.cnf (Master FP Systems)

```
NoWatchDog
Cp          [sys]<sys>CpBtos.R.sys
Include     [sys]<sys>Cp00.R.cnf
Dp          [sys]<sys>DpBtos.R.sys
Include     [sys]<sys>Dp00.R.cnf
Sp          [sys]<sys>SpBtos.R.sys
Tp          [sys]<sys>TpBtos.R.sys
Include     [sys]<sys>Tp00.r.cnf
Ap          [sys]<sys>Centix.R.sys:[sys]<sys>Centix20.R.sys
End
```

[sys]<sys>master.r.cnf (Master DP Systems)

```
NoWatchDog
Cp          [sys]<sys>CpBtos.R.sys
Include     [sys]<sys>Cp00.R.cnf
Tp          [sys]<sys>TpBtos.R.sys
Include     [sys]<sys>Tp00.R.cnf
Ap          [sys]<sys>Centix.R.sys:[sys]<sys>Centix20.R.sys
End
```

[sys]<sys>InitFp00.r.jcl (Master FP Systems with Disk Cartridge)

```
$continueonerror
$run [sys]<sys>TmpFilter.run,[sys]<$00001>
$run [sys]<admin>MCreateSwapR.run
$run [sys]<admin>MCreatePartition.run,150k,UFS
$run [sys]<admin>MInstallServer.run,UFS,[sys]<sys>UFS.run
$run [sys]<sys>MInstallQMgr.run
$run [sys]<sys>MfAdminAgent.run
```

[sys]<sys>InitFp00.r.jcl (Master FP Systems with QIC Tape)

```
$continueonerror
$run [sys]<sys>TmpFilter.run,[sys]<$00001>
$run [sys]<admin>MCreateSwapR.run
$run [sys]<admin>MCreatePartition.run,150k,UFS
$run [sys]<admin>MInstallServer.run,UFS,[sys]<sys>UFS.run
$run [sys]<admin>MCreatePartition.run,60k,Quick
$run [sys]<admin>MInstallServer.run,Quick,
[sys]<admin>MQicServer.run
$run [sys]<sys>MInstallQMgr.run
$run [sys]<sys>MfAdminAgent.run
```

[sys]<sys>InitDp00.r.jcl (Master FP Systems)

```
$run [sys]<admin>MTapeServer.run,2,16
```

[sys]<sys>InitDp00.r.jcl (Master DP Systems)

```
$continueonerror
$run [sys]<sys>TmpFilter.run,[sys]<$00001>
$run [sys]<admin>MCreateSwapR.run
$run [sys]<admin>MCreatePartition.run,150k,UFS
$run [sys]<admin>MInstallServer.run,UFS,[sys]<sys>UFS.run
$run [sys]<admin>MCreatePartition.run,80k,Tape
$run [sys]<admin>MInstallServer.run,Tape,
[sys]<admin>MTapeServer.run,2,16
$run [sys]<sys>MInstallQMgr.run
$run [sys]<sys>MfAdminAgent.run
```

[sys]<sys>InitCp00.r.jcl

```
$run[sys]<sys>MSpoolerMgr.run
$END
```

[sys]<sys>InitSp00.r.jcl

```
$run [sys]<admin>MTapeServer.run,2,16
```

[sys]<sys>InitTp00.r.jcl

```
$END
```

[sys]<sys>Fp00.r.cnf (Master FP Systems)

```
BlockType=y, Number=18
BlockType=z, Number=40
DISK=0   DEVICE=d0,   PASSWORD=d0
DISK=1   DEVICE=d1,   PASSWORD=d1
DISK=2   DEVICE=d2,   PASSWORD=d2
DISK=3   DEVICE=d3,   PASSWORD=d3
```

[sys]<sys>Dp00.r.cnf

```
BlockType=y, Number=18
BlockType=z, Number=40
DISK 0    DEVICE=s0, PASSWORD=s0
DISK 1    DEVICE=s1, PASSWORD=s1
DISK 2    DEVICE=s2, PASSWORD=s2
DISK 3    DEVICE=s3, PASSWORD=s3
DISK 4    DEVICE=s4, PASSWORD=s4
DISK 5    DEVICE=s5, PASSWORD=s5
```

[sys]<sys>Cp00.r.cnf

```
Block Type = x, Number = 16
Block Type = y, Number =  6
Block Type = z, Number = 60
async  3, speed=9600, parity=none, stopbits=1, charbits=8
line1, clock=307k
line2, clock=307k
pt 1
```

[sys]<sys>Sp00.r.cnf

```
Block Type = y, Number =  4
Block Type = z, Number = 30
```

[sys]<sys>ConfigUFS.r.sys (Master FP Systems)

```
%      UFS Configuration File for bootable image master fp

Root    [sys]<sys>partition.R.0      FP00      255
Swap    [sys]<sys>partition.R.1      AP00      254
        [sys]<sys>partition.R.2      FP00      250
        [d1]<sys>partition.0         FP00      128
        [d1]<sys>partition.1         FP00      129
        [d1]<sys>partition.2         FP00      130
        [d1]<sys>partition.3         FP00      131
        [d1]<sys>partition.4         FP00      132
        [d1]<sys>partition.5         FP00      133
        [d1]<sys>partition.6         FP00      134
        [d1]<sys>partition.7         FP00      135
        [d2]<sys>partition.0         FP00      136
        [d2]<sys>partition.1         FP00      137
        [d2]<sys>partition.2         FP00      138
        [d2]<sys>partition.3         FP00      139
        [d2]<sys>partition.4         FP00      140
        [d2]<sys>partition.5         FP00      141
        [d2]<sys>partition.6         FP00      142
        [d2]<sys>partition.7         FP00      143
        [d3]<sys>partition.0         FP00      144
        [d3]<sys>partition.1         FP00      145
        [d3]<sys>partition.2         FP00      146
        [d3]<sys>partition.3         FP00      147
        [d3]<sys>partition.4         FP00      148
        [d3]<sys>partition.5         FP00      149
        [d3]<sys>partition.6         FP00      150
        [d3]<sys>partition.7         FP00      151
Tape    [qic]                                  000
Tape    [tape]                                 001
```

[sys]<sys>ConfigUFS.r.sys (Master DP Systems)

```
%      UFS Configuration File for bootable image master dp

Root   [sys]<sys>partition.R.0      DP00      255
Swap   [sys]<sys>partition.R.1      AP00      254
       [sys]<sys>partition.R.2      DP00      250
       [s0]<sys>partition.0         DP00      128
       [s0]<sys>partition.1         DP00      129
       [s0]<sys>partition.2         DP00      130
       [s0]<sys>partition.3         DP00      131
       [s0]<sys>partition.4         DP00      132
       [s0]<sys>partition.5         DP00      133
       [s0]<sys>partition.6         DP00      134
       [s0]<sys>partition.7         DP00      135
       [s1]<sys>partition.0         DP00      136
       [s1]<sys>partition.1         DP00      137
       [s1]<sys>partition.2         DP00      138
       [s1]<sys>partition.3         DP00      139
       [s1]<sys>partition.4         DP00      140
       [s1]<sys>partition.5         DP00      141
       [s1]<sys>partition.6         DP00      142
       [s1]<sys>partition.7         DP00      143
       [s2]<sys>partition.0         DP00      144
       [s2]<sys>partition.1         DP00      145
       [s2]<sys>partition.2         DP00      146
       [s2]<sys>partition.3         DP00      147
       [s2]<sys>partition.4         DP00      148
       [s2]<sys>partition.5         DP00      149
       [s2]<sys>partition.6         DP00      150
       [s2]<sys>partition.7         DP00      151
Tape   [qic]                                  000
Tape   [tape]                                 001
```

Printer Spooler Configuration Files

[sys]<sys>Queue.index

```
SPL/[sys]<spl>SPL.queue/1/1
SPLB/[sys]<spl>SPLB.queue/1/1
SPOOLERSTATUS/[sys]<spl>SPOOLERSTATUS.queue/1/1
PARALLELcontrol/[sys]<spl>parallelcontrol.queue/1/1
SERIALcontrol/[sys]<spl>serialcontrol.queue/1/1
```

[sys]<sys>SplCnfg.sys

```
0/PARALLEL/SPL/[sys]<spl>SPLconfigCp00.sys/64/y/Cp00
C/SERIAL/SPLB/[sys]<spl>SPLCconfigCp00.sys/64/y/Cp00
```

Disk Device Configuration Files

[sys]<sys>IVSyquest6.cnf:

```
; THIS IS FOR A SYQUEST DRIVE
CYLINDERS=288
TRACKS=2
SECTORS=17
BYTES=512
STEP=0
REMOVABLE=yes
END
```

[sys]<sys>IVAtasi46.cnf:

```
; THIS IS FOR AN ATASI 46 MB DRIVE
CYLINDERS=645
TRACKS=7
SECTORS=16
BYTES=512
STEP=0
REMOVABLE=no
END
```

5/12

[sys]<sys>IVMicropolis85.cnf:

```
; THIS IS FOR A MICROPOLIS 85MB DRIVE
CYLINDERS=1024
TRACKS=8
SECTORS=17
BYTES=512
STEP=0
REMOVABLE=no
END
```

[sys]<sys>IVToshiba85.cnf:

```
; THIS IS FOR A TOSHIBA 85 MB DRIVE
CYLINDERS=830
TRACKS=10
SECTORS=17
BYTES=512
STEP=0
REMOVABLE=no
END
```

[sys]<sys>IVMemorex166.cnf:

```
; THIS IS FOR A MEMOREX 166MB DRIVE
CYLINDERS=823
TRACKS=10
SECTORS=32
REMOVABLE=no
END
```

Hardware Configuration Information

Figure C-1 **Built-in Disk Device Naming Conventions**

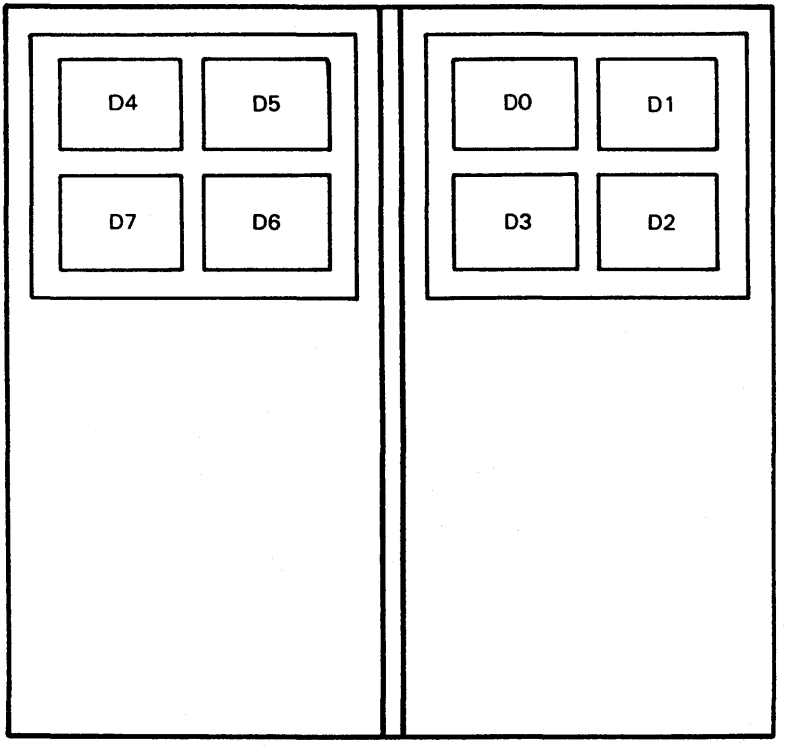
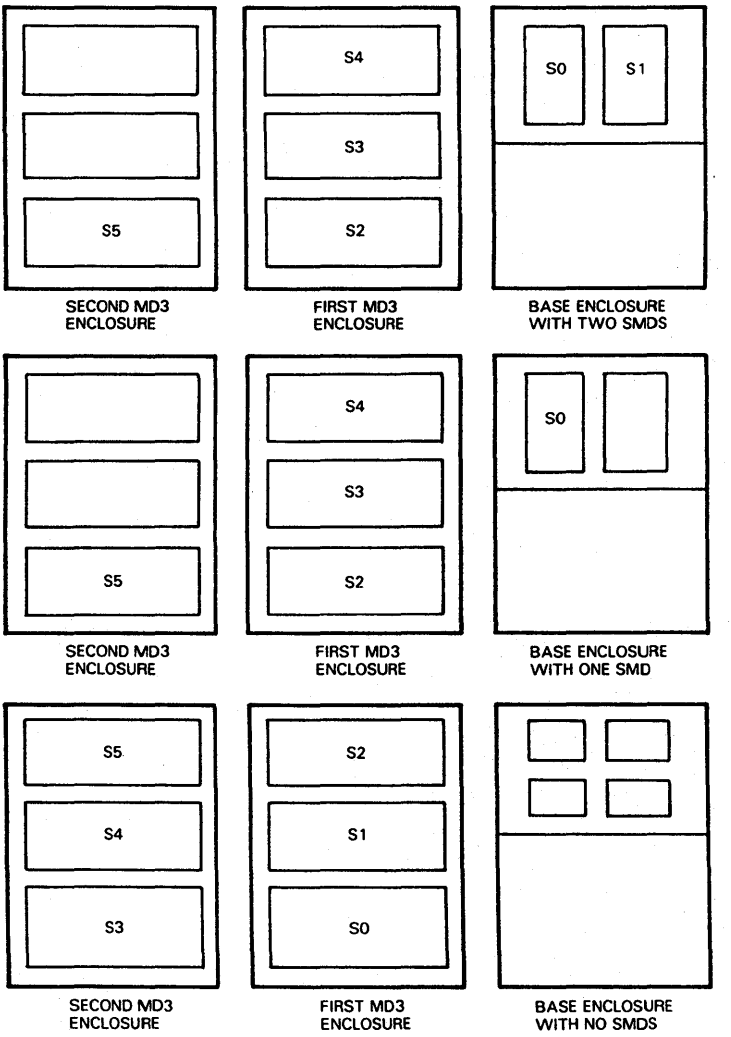


Figure C-2 **SMD Disk Device Naming Conventions**



ALL VIEWS ARE FROM THE REAR OF THE ENCLOSURES.

E7560

Figure C-3 **Processor Board Numbering Scheme**

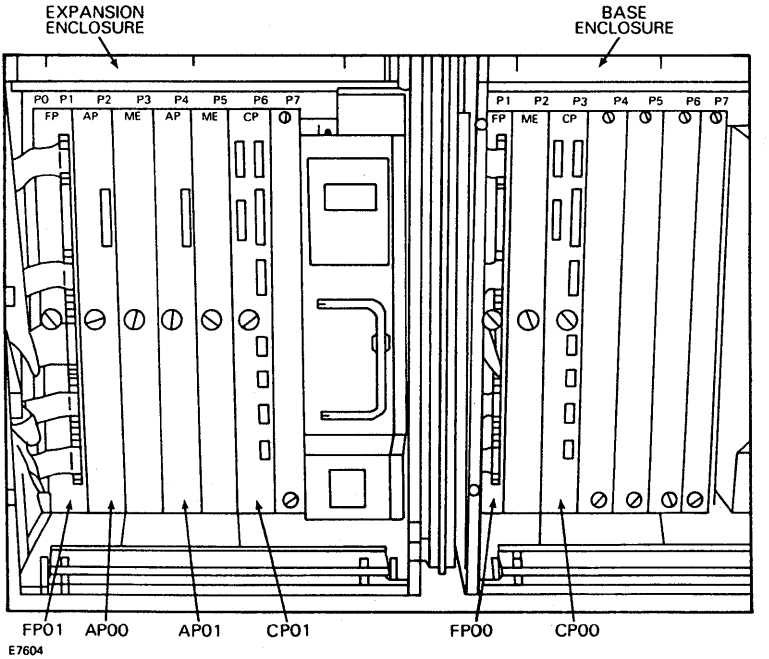


Table C-1 **Processor Slot Numbering Scheme**

Enclosure	P0	P1	P2	P3	P4	P5	P6	P7
Base	70	71	72	73	74	75	76	77
Expansion 1	60	61	62	63	64	65	66	67
Expansion 2	50	51	52	53	54	55	56	57
Expansion 3	40	41	42	43	44	45	46	47
Exapnsion 4	30	31	32	33	34	35	36	37
Expansion 5	20	21	22	23	24	25	26	27

New CENTIX Device Naming Conventions

There are new CENTIX device naming conventions for the 6.0 release of the system software.

For the 6.0 release, the old device names are linked to the new device names, so you can continue using the old device names. However, if you plan on acquiring future releases of CENTIX system software, it is recommended that you begin converting to the new device naming convention.

Table D-1 lists the naming conventions for built-in disk devices.

Table D-2 lists the naming conventions for SMD disk devices.

Table D-3 lists the naming conventions for tape devices.

Note: *In the disk names in Tables D-1 and D-2,* n *represents the partition number. Each disk can have up to eight partitions (0 through 7).*

Table D-1 **Naming Conventions for Built-In Disk Drives**

Pre-6.0 Release	6.0 Release and Later	BTOS Disk Device Name
FIRST FP		
/dev/[r]fp00*n*	/dev/[r]dsk/c0d00s*n*	d0 (disk cartridge)
/dev/[r]fp01*n*	/dev/[r]dsk/c0d01s*n*	d1
/dev/[r]fp02*n*	/dev/[r]dsk/c0d02s*n*	d2
/dev/[r]fp03*n*	/dev/[r]dsk/c0d03s*n*	d3
SECOND FP		
/dev/[r]fp04*n*	/dev/[r]dsk/c0d04s*n*	d4
/dev/[r]fp05*n*	/dev/[r]dsk/c0d05s*n*	d5
/dev/[r]fp06*n*	/dev/[r]dsk/c0d06s*n*	d6
/dev/[r]fp07*n*	/dev/[r]dsk/c0d07s*n*	d7
THIRD FP		
/dev/[r]fp08*n*	/dev/[r]dsk/c0d08s*n*	d8
/dev/[r]fp09*n*	/dev/[r]dsk/c0d09s*n*	d9
/dev/[r]fp10*n*	/dev/[r]dsk/c0d10s*n*	d10
/dev/[r]fp11*n*	/dev/[r]dsk/c0d11s*n*	d11
and so on		

Table D-2 **Naming Conventions for SMD Disk Drives.**

Pre-6.0 Release	6.0 Release and Later	BTOS Disk Device Name
FIRST DP		
/dev/[r]dp00*n*	/dev/[r]dsk/c1d00s*n*	s0
/dev/[r]dp01*n*	/dev/[r]dsk/c1d01s*n*	s1
/dev/[r]dp02*n*	/dev/[r]dsk/c1d02s*n*	s2
/dev/[r]dp03*n*	/dev/[r]dsk/c1d03s*n*	s3
/dev/[r]dp04*n*	/dev/[r]dsk/c1d04s*n*	s4
/dev/[r]dp05*n*	/dev/[r]dsk/c1d05s*n*	s5
SECOND DP		
/dev/[r]dp06*n*	/dev/[r]dsk/c1d06s*n*	s6
/dev/[r]dp07*n*	/dev/[r]dsk/c1d07s*n*	s7
/dev/[r]dp08*n*	/dev/[r]dsk/c1d08s*n*	s8
/dev/[r]dp09*n*	/dev/[r]dsk/c1d09s*n*	s9
/dev/[r]dp10*n*	/dev/[r]dsk/c1d10s*n*	s10
/dev/[r]dp11*n*	/dev/[r]dsk/c1d11s*n*	s11
THIRD DP		
/dev/[r]dp12*n*	/dev/[r]dsk/c1d12s*n*	s12
/dev/[r]dp13*n*	/dev/[r]dsk/c1d13s*n*	s13
/dev/[r]dp14*n*	/dev/[r]dsk/c1d14s*n*	s14
/dev/[r]dp15*n*	/dev/[r]dsk/c1d15s*n*	s15
/dev/[r]dp16*n*	/dev/[r]dsk/c1d16s*n*	s16
/dev/[r]dp17*n*	/dev/[r]dsk/c1d17s*n*	s17
and so on		

Table D-3 **Naming Conventions for Tape Drives.**

	Pre-6.0 Release	6.0 Release and Later
first half-inch drive	/dev/rmt0	/dev/rmt/c1d0
QIC drive	/dev/rmt1	/dev/rmt/c0d0
second half-inch drive	/dev/rmt2	/dev/rmt/c1d1
third half-inch drive	/dev/rmt3	/dev/rmt/c1d2
and so on		

Glossary

AdminAgent. The system service that manages the execution of the master utilities on XE 500 processors. The AdminAgent can be run in either single mode or multiple mode.

AP. See **Applications Processor.**

Applications Processor (AP). Processor board in the XE 500 system that runs the CENTIX operating system.

archive. The procedure used to backup system files onto portable media.

archive volume. A portable medium that is used to archive (back up) files.

ASCII (American National Standard Code for Information Interchange). Control and graphic character set consisting of 7-bit coded characters (8 bits including parity check), used for information interchange between data communications systems.

background process. Process that, once started up, runs underneath any active processes, with no interaction with the user through the terminal.

bad spot. Any part of a hard disk drive that is faulty and cannot record information.

Batch utility. A utility that provides a job stream capability and allows programs to be running in a processor's secondary partitions while a user-interactive program runs in a primary partition.

B 20. Burroughs microcomputer from the clustered workstation series.

block. On a disk device, a 512-byte subdivision of data on the disk. Also referred to as a *sector.*

block device. A hardware device that handles I/O data in 1024 bytes (1 kB) blocks. The I/O size is controlled by the operating system's buffer size and is independent of the user's I/O size. Disk and tape devices can be configured as block devices.

BootLoadParams file. A file that defines how the system disk is to be initialized during Boot Load.

Boot Load utility log. A log that contains a record of the Boot Load session in the file [sys]<sys>BootLoad.log.

bootup. The operation that starts up the system when the keyswitch on the XE 500 base enclosure is turned from STOP to MANUAL, REMOTE, or NORMAL. The bootup operation includes running a series of self-diagnostic tests, loading the master operating system, and reading the set of system files associated with the position to which the keyswitch is turned.

Bourne Shell. A command-oriented shell used to search for programs in specific places on the CENTIX file system.

BTOS. B 20 Operating System. All XE 500 boards except the Applications Processor run a version of BTOS. BTOS-based processors handle all of the actual data transfers for CENTIX between the XE 500 and I/O devices (such as disk drives, tape drives, terminals, and modems).

BTOS Executive. See **Executive**.

BTOS status codes. Numerical codes reported by the system software through the BTOS Executive and other facilities. These codes indicate the status of system operations.

BTOS workstation. A Burroughs model B 21, B 22, B 26, B 27, or B 28 workstation.

built-in disk device. A 37.5, 71.3, or 72.2 MB, 5 1/4-inch hard disk drive, or a 135-MB SMD disk device, housed in an XE 500 or MD3 enclosure and used for the mass storage of programs and data.

CENTIX. Burroughs version of the System V UNIX operating system.

CENTIX file system server. Interface between CENTIX I/O requests to a logical disk device and the actual BTOS-based disk I/O operation. The server makes the connection between BTOS partitions and the CENTIX file systems that reside on those partitions.

CentixLoad utility. A menu-driven facility used to install CENTIX system software and optional products into the XE 500 system.

CENTIX shell. Command interpreter; program acting as interface between operating system and users.

central processing unit (CPU). In a computer, the unit that performs most of the fundamental processing. In the XE 500 system, this term is sometimes used interchangeably with the term *processor.*

centrEASE. A menu-driven, interactive facility that you can use to perform many administrative tasks on the CENTIX system.

character device. A hardware device that handles raw data streams. The size of I/O transfers in raw data streams is determined either by the software design of the device itself (for terminals and printers) or by the program controlling the device (for disks and tapes).

CLI. See **Command Line Interpreter.**

cluster line. The RS-422 cluster communications port connection which allows workstations to communicate with the XE 500.

Cluster Processor (CP). Board in XE 500 system; runs communications software and supports PT 1500 terminals, B 20 workstations, a parallel printer, and up to three RS-232-C serial devices.

command. A BTOS command is the command name and command form used with the BTOS Executive. Each command invoked from the Executive causes a corresponding utility to run on a BTOS processor.

command file. A file created by the system administrator which contains a listing of the BTOS commands available to a specific user. The command file allows the system administrator to limit a user's access to selected commands.

command form. The interactive form associated with many BTOS commands in which a user specifies parameters that define how the command is to be executed.

Command Line Interpreter (CLI). An interface program that interprets statements and executes BTOS run files on XE 500 processors.

console. The terminal designated by the system software for use by the system administrator.

CP. See **Cluster Processor.**

CPU. See **central processing unit.**

crash dump. A copy of the memory image of a processor at the time of a nonrecoverable failure (crash). Such a file can be used by Burroughs personnel to analyze the cause of a processor failure.

CRC. See **Cyclical Redundancy Check.**

customized mode. The state of the system when it is booted up using a customized set of system configuration files.

Customizer. In BTOS, a utility that allows a user to modify operating system parameters to make the operating system configuration better-suited to the user's needs.

Cyclical Redundancy Check (CRC). A method of data error detection in which a two-byte block check character that is appended to the data block. This operation insures that the data is not corrupted and that corrupted data is detected. See also Error Correction Code.

cylinder. A set of tracks on a storage device that can be accessed together as a unit.

DAM. See **Direct Access Method.**

data communication (data comm). The transfer of data between a data source and data link using one or more data links according to the designated protocol.

data volume. A volume that is used for data and program storage.

debugger. A program that aides in fixing errors (bugs) in programs. It allows you to examine and modify memory, to set and clear breakpoints, and to produce formatted displays of memory.

demand paging. A form of memory management that keeps in on-board memory only those parts of the program code and data required for execution.

device. A terminal, printer, disk, tape, or other input/output medium that can be attached to the system. A device can be physical or logical.

device file. In the CENTIX file system, a file in the /dev directory that represents a terminal, printer, disk, tape, or other input/output device.

Direct Access Method (DAM). A file access method providing random and non-overlapped input or output of records.

directory. In BTOS, a related group of files on a volume. Directories are the second level of the BTOS three-level hiearchical file system. In CENTIX, a directory is a list of files that are assigned to the directory. A directory can also contain other directories.

disk cartridge. Magnetic disk storage medium utilizing a hard disk enclosed in a portable cartridge. Disk cartridges are used with a disk cartridge drive of an XE 500 base enclosure.

disk device. A hardware component that is used to store data. An XE 500 disk device may be either a disk cartridge, a built-in disk (that is, a 5 1/4-inch hard disk controlled by a File Processor), or a Storage Module Device (SMD) disk.

disk extent. One or more contiguous disk sectors that contain all or part of a file.

disk initialization. See **volume initialization.**

Disk Processor (DP). Processor board in an XE 500 system that is formed by connecting SC to SP. The DP supports I/O to half-inch magnetic tape drives and MD3 disks.

DP. See **Disk Processor.**

dumb terminal. Unprogrammable terminal that uses ASCII code.

ECC. See **Error Correction Code.**

Error Correction Code (ECC). A method of detecting and correcting data errors. ECC is a four-byte block check character appended to the data block that allows corrupted data bits to be identified and corrected. In a manner similar to CRC, it initiates error detection but also allows for correction operations on data that is read from a disk.

/etc/getty. Process that readies terminal connections for login.

/etc/gettydefs. File that contains line communications information on terminals. It is read by the /etc/getty process.

/etc/inittab*nn*. CENTIX files that defines the CENTIX terminal assignments and the CENTIX processes that are started at boot time.

/etc/rc. File used as a general purpose start-up file for various background processes such as the lp spooler scheduler and cron.

Executive. An interactive BTOS application program through which the various BTOS utilities can be executed through command forms.

File Processor (FP). Processor board in an XE 500 system that supports I/O operations to disk devices.

file system. A collection of files that are all stored on the same logical disk device. In CENTIX, a file system must be attached to, or is "subordinate to," a directory. The file system physically contains the files that are logically contained in that directory. The term can also be used, as in "the CENTIX file system," to describe the entire hierarchy of directories, specific file systems, and files in a CENTIX system. A BTOS file system is made up of a three-level hierarchy that includes a volume, directories, and the files contained within each directory.

FP. See **File Processor.**

hexadecimal values. The notation system of representing numbers in base 16.

home directory. For a user, the directory into which the user is automatically placed when he or she logs onto the system.

ICC. See **Inter-CPU Communication.**

Indexed Sequential Access Method (ISAM). Programming tool that uses an index to sequence file records on disk and to access those records directly.

inode. In a CENTIX file system, there is one inode for each file and directory in the file system. The inode contains status information for its file or directory, such as the size, its owner and permissions, its disk address list, and whether it is a directory, an ordinary file, or a special file.

installation logs. A log that provides a record of software installation operations.

Inter-CPU Communication (ICC). Intercommunications protocol used by XE 500 processors when communicating across the system bus.

I/O. Input/output.

I/O device configuration files. File that define the hardware and software parameters of disk drives, printers, tape drives, and communications devices such as modems and RS-232-C serial terminals.

ISAM. See **Indexed Sequential Access Method.**

ISAM server. The application that manages the execution of ISAM-related operations on a BTOS processor.

JCL. See **Job Control Language.**

Job Control Language (JCL). Statements submitted to CLI for batch execution.

kb. See kilobit.

kB. See kilobyte.

kernel. Portion of the CENTIX operating system that controls system processes and allocates system resources.

kilobit. 1024 bits.

kilobyte. 1024 bytes.

language development software. The software tools that support BTOS programming languages.

Large-Scale Integration (LSI). Monolithic integrated circuits of very high density.

log file. In BTOS, a file which contains a record of operations performed by BTOS processors.

LSI. See **Large-Scale Integration.**

master commands. A set of BTOS commands, executed from the Executive, which invoke the corresponding utilities that run on an XE 500 processor. See **utilities.**

master configuration file. A configuration file used by the master processor (FP00 or DP00) to determine the processor operating systems to be loaded at boot time.

master processor. The first processor in the XE 500 system. This processor is loaded with the master BTOS operating system when the system is started. The master processor then controls the loading of the other processors with their operating systems. It also manages the system's interface with the base enclosure front panel controls.

Master Utilities. Utilities that are invoked when the BTOS Master Commands are used.

MB. See **megabyte.**

MBTOS Config utility. A BTOS utility that allows a user to create or modify the master configuration file, processor initialization files, and processor configuration files.

MCommands. See **master commands.**

MD3. See **Multidisk 3.**

ME. See **Memory Expansion Board.**

megabyte. 1,024,000 bytes.

Memory Expansion Board (ME). Board attached to a processor board to supply 1/2 M-byte or 1 M-byte additional memory capacity.

Memory Management Unit (MMU). Part of the AP that supports multiprogramming and demand paging.

menu. In BTOS, a terminal screen listing of operations from which a user selects to execute the desired operation.

mixed system. An XE 500 system that contains a complete BTOS operating system and a complete CENTIX operating system.

MMU. See **Memory Management Unit.**

MSysload utility log. A log containing a record of the MSysload session. It appears in the file [sys]<sys>SysLoad.log.

Multidisk 3 (MD3). A freestanding enclosure supporting up to three 135-MB (formatted) SMD disk drives.

multiuser mode. An operating state defined in the /etc/inittab*nn* files. In multiuser mode, user terminals are readied for login.

MUtilities. See **Master Utilities.**

normal mode. An XE 500 operating mode that exists when the keyswitch is turned to NORMAL and the system is booted up using the default set of system files.

null string. A sequence of blank characters that usually does not change the default condition set up in the original BTOS command or submit file.

ofcli. CENTIX command used to access the BTOS Command Line Interpreter (CLI) mode, from which BTOS MCommands can be initiated. See **Command Line Interpreter.**

operating mode. The XE 500's operating mode is determined by any one of four sets of system configuration files and BTOS processor versions used to boot up the XE 500. The particular set of system configuration files used during boot-up time depends on the keyswitch position.

operating system. The part of the system software that supervises the running of individual programs. Its functions include loading programs, allowing concurrent operation of two of more programs, scheduling processes within the system, and providing management of information. The operating system is sometimes referred to as the *system image.*

partition. An assigned area of the XE 500 BTOS processor's memory. A program being run by the processor can be assigned a partition in which to execute. The program restricts its instruction and data to this assigned area of memory. This allows the processor to operate in a multitasking mode, in which several programs are running on the processor at the same time. Each program is actually sharing processing time on the processor. Each partition is associated with a CENTIX logical disk device. See **primary partition** and **secondary partition.**

password. A string of characters used as part of of the BTOS file security system. A password can be assigned to a user, device, volume, directory, or file. Once passwords are assigned (at least a volume and directory password) and a protection level is assigned to a file, a user must enter the correct password to gain access to that file, depending on the protection level.

path name. For a CENTIX file, the name that identifies the file's position in the CENTIX file system. A complete (absolute) path name always begins with /, which stands for the root directory.

Pilf factor. A value that can be specified when you create a file system to control the size of the blocks of data that are moved in and out of the file system in I/O operations.

piping. Linking of programs so that the output of one program becomes the input for another program.

port. The part of a data processor dedicated to a single data channel for receiving data from, or transmitting data to, one or more external remote devices.

primary partition. An assigned area of an XE 500 processor's memory that is established during system start up. Services that require user interaction and some system services must be run in the primary partition. There is only one primary partition per processor. See **partition.**

printer spooler. A utility that allows printing tasks to be queued on a disk device and then automatically sent to an available printer, thus allowing a user to execute other applications from his or her terminal.

printer spooler manager. A system service that manages the transfer of data from disk files to printers.

printer translation file. In BTOS, a file that is used to translate characters in a text file into different characters or printer control sequences as the text file is printed.

processor configuration file. A file that defines the hardware and software configuration associated with a processor board.

processor initialization file. A file that defines the system services that are to be run on a processor when the system is started up.

product content file. A file containing entries for each new software product that is stored on a release disk cartridge. This file provides the Boot Load and MSysload utilities with instructions on how to load the product.

protection level. A decimal value assigned to a file that determines the type of password, if any, that a user must enter to gain read or write access to that file.

QIC. See **Quarter-Inch Cartridge Tape.**

QIC server. An application that controls the execution of I/O operations to a QIC tape drive.

Quarter-Inch Cartridge (QIC) Tape. Magnetic tape storage medium that utilizes quarter-inch-wide tape enclosed in a portable cartridge. QIC tapes are used with a disk cartridge drive of an XE 500 base enclosure.

queue index file. A file that defines the queues for which the Queue Manager is responsible.

Queue Manager. A BTOS system service that controls the various queues in which related tasks are stored for processing. For example, the Queue Manager controls the queues that are used to handle print requests made through printer spoolers. It can also manage similar queues for data communications processes and user-defined processes.

RAM. See **random-access memory.**

random-access memory (RAM). The onboard processor memory to which and from which data can be written and read nonsequentially.

raw device. A block device configured to accept raw data. The size of I/O transfers for a raw data stream is determined by the design of the device itself or by the program controlling the device.

read-only memory (ROM). The onboard processor memory whose contents can be read but can be written only by special means. The ROM normally contains start-up programs, such as self-diagnostic tests.

register. A temporary memory location for data.

release medium. In BTOS, a disk cartridge, QIC tape, or half-inch tape on which Burroughs release software is stored.

restore. The procedure by which archive files are written from an archive medium to an XE 500 volume.

restricted mode. An XE 500 operating mode that is determined by the keyswitch setting of the XE 500 base enclosure. When set to REMOTE, the system reads the set of system configuration files that contain the identifying character ".r".

ROM. See **read-only memory.**

root. The base directory of the CENTIX file system. Every CENTIX directory must either be subordinate to root, or subordinate to a directory that is subordinate to root, or subordinate to a directory that is subordinate to a directory that is subordinate to root, and so on. In a file path name, root is represented by a slash (/).

Root/Swap utility. An interactive utility used to create the root partition and swap file for APOO. (See **Applications Processor.**)

RS-232-C. A designation that refers to an industry specification developed to standardize the interface between different types of communications equipment.

RS-422. A high-speed communications standard used in an XE 500 system to link cluster workstations.

run statement. A line of text that can be included in a JCL file for execution on a processor by the CLI.

Run-Time System. CENTIX shell commands and software to support an office environment running office application programs.

saf. Command entered that initiates the centrEASE administrative facility.

SCCS. See **Source Code Control System.**

sector. See **block.**

secondary partition. An assigned area of an XE 500 processor's memory in which applications, including some system services, that do not require user interaction can be run.

shell. The portion of the CENTIX operating system that provides a user interface to the kernel.

shell script. An executable CENTIX file that contains a program comprised of shell commands.

sign-on file. A required file for each user to be allowed to log into the system. Among other things, this file defines the user's password, the BTOS commands the user can access, the path (volume and directory) a user will be placed after signing on, and the application a user will enter.

single user mode. An operating state defined in the /etc/inittab*nn* files. In single user mode, only the system console can access the system.

SMD. See **Storage Module Device.**

SNA. See **Systems Network Architecture.**

Sort/Merge. Programming tool that provides sequencing of file records and merging of sorted records from more than one file.

Source Code Control System (SCCS). A group of software commands that control and account for changes to text files.

SP. See **Storage Processor.**

special file. See **device file.**

spooler configuration file. A file containing information that the system uses to coordinate print operations between the spooler and the queue manager.

status codes. See **BTOS status codes.**

STATUS display. The display on the XE 500 base enclosure's front panel through which the status of software installation, system start-up operations, and error conditions are reported.

Storage Controller (SC). Board used with SP to form DP, which controls SMDs.

Storage Module Device (SMD). 132-byte (formatted) Memorex 166 Disk Drive.

Storage Processor (SP). Processor board in XE 500 system; controls half-inch magnetic tape.

subcommand. A subcommand appears within the operational procedures of a command and makes available an additional operation.

submit file. In BTOS, a file that causes a series of BTOS utilities to be executed.

superblock. The portion of a CENTIX file system that contains descriptions of the file system, including the file system name, its size in blocks, the number of blocks reserved for inodes, the free inode list, and the free block list.

superuser. The name by which the system administrator is called in CENTIX documentation. To become superuser, the administrator signs onto the system as "root".

surface test. A disk initialization operation that checks for bad spots on a disk that is accomplished by writing data to and then reading the data from each sector of the disk.

swap area. Portion of the XE 500 disk used to temporarily store jobs that are not currently being processed.

system bus. Path over which the system processors communicate.

system configuration files. Text files that define the XE 500 BTOS hardware and software configuration.

system crash status words. Information normally displayed at the terminal screen and entered in the system log after a system crash occurs. System crash status words are useful when isolating the cause of a system crash.

system disk. In BTOS, the disk on which the XE 500 system software is installed. The system disk is normally either built-in disk d1 (for systems with a master FP) or SMD disk s0 (for systems with a master DP).

system image. The run file for an operating system.

system log. A log, displayable through the **MPLog** command, that contains a detailed account of system status, errors, and failures.

Systems Network Architecture (SNA). A formal set of rules for designing, building, and operating the components of a data communications network.

tape server. An application that controls the execution of I/O operations to a half-inch tape drive.

Temporary Directory Filter. In XE 500 BTOS, a system service that causes all temporary files, regardless of what application created them, to be stored in one directory on an XE 500 volume.

terminal. A device, usually equipped with a keyboard and a display, which is capable of sending and receiving information over a communication channel.

Terminal Processor (TP). Processor board in XE 500 system that supports a parallel printer and up to ten RS-232-C serial devices.

TP. See **Terminal Processor.**

UNIX. AT&T Bell Laboratories operating system designed for application program development on various computer systems.

user command file. In BTOS, a file that contains all of the BTOS commands to which a user may have access. A command file entry in a user's signon file defines the command file accessible to the user.

utility. A BTOS utility is a program that may be invoked by executing a command from the Executive or by executing a utility's run file through the Command Line Interpreter (CLI) using Job Control Language (JCL).

VHB. See **Volume Home Block.**

volume. In BTOS, the complete file system unit of information stored on a formatted disk.

volume fragmentation. The scattering of disk extents available for data storage. A volume is said to be fragmented if the system must allocate two or more smaller disk extents that have a total size sufficient to fulfill a data storage request. See **disk extent.**

Volume Home Block (VHB). An area on a volume that contains the volume name, the date it was created, the date it was last modified, and the number of free pages and file headers.

volume initialization. The procedure performed by the MIVolume utility in which an XE 500 disk is prepared for use as a system volume. Among other tasks, this procedure formats the disk surfaces, performs read/write tests, writes volume control structures, and creates system files.

x, y, z blocks. Operating system parameters that control the number and size of cluster communication blocks.

wild card character. The BTOS system allows two wild card characters to be used in command form fields: an asterisk (*) and a question mark (?). The asterisk represents any string of characters; the question mark represents any individual character. Some operations allow you to use wild card characters in file specifications. The system then tries to match the portion of the name that appears before or after the wild card character and performs the requested operation for each matched file name.

Index

A

B

M

R

S

T

U

V

X

Title: _____

Form Number: _____ Date: _____

Burroughs Corporation is interested in your comments and suggestions regarding this manual. We will use them to improve the quality of your Product Information.

Please check type of suggestion: ☐ Addition ☐ Deletion ☐ Revision ☐ Error

Comments: _____

_____

_____

_____

Name _____

Title _____

Company _____

Address _____
Street City State Zip

Telephone Number ( ) _____
Area Code

Title: _____

Form Number: _____ Date: _____

Burroughs Corporation is interested in your comments and suggestions regarding this manual. We will use them to improve the quality of your Product Information.

Please check type of suggestion: ☐ Addition ☐ Deletion ☐ Revision ☐ Error

Comments: _____

_____

_____

_____

Name _____

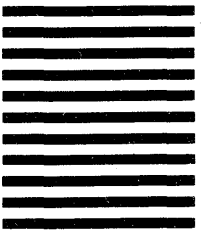
Title _____

Company _____

Address _____
Street City State Zip

Telephone Number ( ) _____
Area Code

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES



BUSINESS REPLY CARD

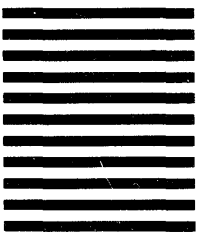
FIRST CLASS PERMIT NO. 817 DETROIT, MI 48232

POSTAGE WILL BE PAID BY ADDRESSEE

Burroughs Corporation
Production Services – East
209 W. Lancaster Avenue
Paoli, Pa 19301 USA

ATTN: Corporate Product Information

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES



BUSINESS REPLY CARD

FIRST CLASS PERMIT NO. 817 DETROIT, MI 48232

POSTAGE WILL BE PAID BY ADDRESSEE

Burroughs Corporation
Production Services – East
209 W. Lancaster Avenue
Paoli, Pa 19301 USA

ATTN: Corporate Product Information