UNISYS

XE500 BTOS

Installation and Implementation Guide

Relative to Release Level 7.0

Priced Item

May 1988
Distribution Code SA
Printed in U S America
5028632

UNISYS

XE500 BTOS

Installation and Implementation Guide

Relative to Release Level 7.0

Priced Item

May 1988
Distribution Code SA
Printed in U S America
5028632

The names, places and/or events used in this publication are not intended to correspond to any individual, group, or association existing, living or otherwise. Any similarity or likeness of the names, places, and/or events with the names of any individual living or otherwise, or that of any group or association is purely coincidental and unintentional.

NO WARRANTIES OF ANY NATURE ARE EXTENDED BY THE 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 in this document 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, rules, 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.

Comments or suggestions regarding this document should be submitted on a Field Communication Form (FCF) with the CLASS specified as 2 (S.S.W.:System Software), the Type specified as 1 (F.T.R.), and the product specified as the 7-digit form number of the manual (for example, 5028632).

About This Guide

This guide contains general information about the installation and the implementation of the software for the XE520 Shared Resource Processor.

Who Should Use This Guide

This guide is intended for the person who installs, configures, and maintains the XE520 software. Some knowledge of computer operations is necessary.

How to Use This Guide

You do not need to read this guide from cover to cover. It contains information on many aspects of the XE520, and you may be interested in only some of them at this time.

If you are familiarizing yourself with XEBTOS installation, configuring, and administration requirements, you should read the brief overview in section 1.

In addition, if you look over the table of contents to review the topics before you start, you may find this guide easier to use.

To locate detailed information, refer to the alphabetic index of topics at the back of this guide.

How This Guide is Arranged

The material in this guide is divided into 12 sections, 2 appendixes, a table of contents, a glossary of terms, and an index.

- Section 1 gives an overview of the XE520 BTOS software installation procedure, including a description of the release media.
- Section 2 describes the XE520 hardware, including the base enclosure and expansion enclosure, and keyswitch positions. It also includes information on powering up the XE520, booting a BTOS workstation from the XE520, and powering down the XE520.

- Section 3 describes how to use quarter-inch cartridge (QIC) tapes and half-inch tapes, and their tape drives.
- Section 4 gives detailed procedures for installing the XEBTOS system software on the XE520.
- Section 5 gives detailed procedures for installing the BTOS standard software on the XE520.
- Section 6 gives an overview of the system configuration files.
- Section 7 briefly describes system services, such as the AdminAgent, tape servers, and QIC servers.
- Section 8 describes how to configure system services and configuration files for the XE520.
- Section 9 discusses post-configuration tasks that you complete after creating the system configuration files. These tasks include creating BTOS command forms and files, obtaining a list of bad spots, restoring customized files, and backing up your system software.
- Section 10 describes how to install and configure the Queue Manager.
- Section 11 describes how to install, configure, and operate the printer spooler.
- Section 12 describes how to create I/O Device Configuration Files, and includes information about the mCREATE CONFIGURATION FILE utility.
- Appendix A lists the default system configuration files.
- Appendix B provides hardware configuration information.

Conventions

The following conventions apply to this guide:

- Processor board names refer to both the standard board and the X-Board. For example, CP refers to both CP and CP-X.
- The term "BTOS workstations" refers to the Unisys B26, B27, B28, and B38 workstations. B21 and B22 workstations are no longer supported.

Related Product Information

Refer to the following product information for more information about the XE520 or BTOS.

- *XE520 System Capabilities Overview*

  This capabilities overview provides a brief description of the features and capabilities of the XE520 System Shared Resource Processor. It is directed toward the user who wants an introduction to the XE520 system.

- *XE500 BTOS Administration Guide*

  This guide provides administration information for the XE520 BTOS system, including using master utilities, verifying and initializing disks, managing XE520 file systems, establishing system and file security, using the command line interpreter, monitoring XE520 processor activity, troubleshooting, and configuring hardware.

- *BTOS II Standard Software Operations Guide*

  This guide contains introductory, procedural, and reference information for using the standard features of XE520 and workstation BTOS. It includes software installation procedures, system configuration instructions, and explanations of standard Executive commands.

- *BTOS II Standard Software Operations Quick Reference Guide*

  This guide is for experienced operators and administrators who are already familiar with the material presented in the *BTOS II Standard Software Operations Guide*. It provides an alphabetical listing of the XE520 and workstation BTOS commands and their command forms.

- *XE500 BTOS Operations Guide*

  This guide provides information on performing those XE520 BTOS tasks that are routinely performed by anyone using the XE520 system.

- *BTOS II Customizer Programming Guide*

  This guide describes how to create customized versions of XE520 BTOS and workstation BTOS.

- *XE500 BTOS Debugger Operations Guide*

  This guide describes how to use the XE520 BTOS Debugger.

- *BTOS II System Status Codes Reference Manual*

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

Contents

Figures

Tables

Overview of the XE520 BTOS Software Installation

To install the XEBTOS software, you perform two major tasks in the following order:

1 You load the XE520 BTOS software onto the XE520 system disk.
2 You load the workstation BTOS software from floppy disks onto the XE520 system disk using workstation software installation procedures.

To implement the software, you perform two major tasks in the following order:

1 You create system configuration files using the mBTOS CONFIG utility (refer to section 8).
2 You perform the necessary post–configuration tasks such as backing up your system files, creating BTOS master command sets, configuring the AdminAgent, configuring the Queue Manager, configuring printer spooler files, and creating input/output (I/O) device configuration files.

XE520 Software Release Media

The XE520 BTOS system software is available on the following types of release media:

- half–inch tapes
- quarter–inch cartridge (QIC) tapes

If your XE520 has an FP master processor, you use the XE520 BTOS system software that is distributed on QIC tape. If your XE520 has a DP master processor, you use the XE520 BTOS system software that is distributed on QIC or half–inch tape.

In addition, you should also have a BTOS workstation software release package on floppy diskettes. You use the diskettes to load the workstation system and environmental software.

Processor Naming Conventions

XE520 processor names take the form Zpnn, where **Zp** is the processor type mnemonic and **nn** is the number of that type in the system. Table 1–1 lists the processor types and their two character mnemonic. Appendix C shows the processor board numbering scheme.

Table 1–1 **Processor Board Designations**

Mnemonic	Processor Type
FP or FP–X	File Processor
DP or DP–X	Disk Processor
CP or CP–X	Cluster Processor
SP or SP–X	Storage Processor
TP or TP–X	Terminal Processor

Note: Depending on the model, a processor board can have 256 or 768 Kb of RAM memory. You can increase the memory of a processor with 256 Kb of RAM memory to 768 Kb by connecting a 512 Kb Memory Expansion (ME) board to it.

Although the standard processor boards with 256 Kb of RAM memory are still supported for release 7.0, Unisys now provides X–Boards, which contain 768 Kb of RAM memory directly on the processor board.

Each XE520 BTOS system has a master processor, which is always the first processor in the base enclosure. The system loads this processor with the master BTOS operating system when you start the system. 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.

Starting Up the System

This section describes the following tasks:

- using the XE520 front panel controls
- powering up and powering down the XE520 enclosures
- booting the XE520
- booting a BTOS workstation from the XE520

The discussion of the start–up procedure assumes that you know how to power up and power down any Unisys BTOS–based workstations that are part of the system. It also assumes that you know how to use floppy disks with the workstation floppy disk drives. If you are unfamiliar with the operation of these workstations, you should read through your Standard Software documentation before proceeding.

XE520 Controls

All of the XE520 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
- QIC tape drive slot (optional)

The keyswitch setting determines the operating mode of the XE520. You use the control panel keys that were packed with the XE520 to turn the keyswitch. You select the operating mode by turning the key to the desired position. You lock the keyswitch at a selected position by removing the key.

The keyswitch positions are defined in table 2–1.

Figure 2–1 **Sample XE520 Base Enclosure Front Panel**

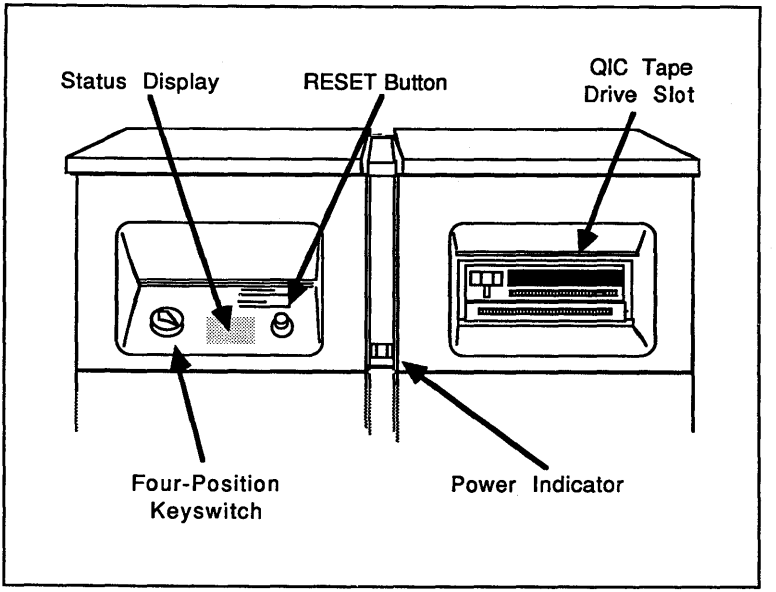


Table 2–1 **Keyswitch Positions on the XE520**

Keyswitch Position	XE520 Operating Mode
STOP	STOP places the system in a reset state and prevents anyone from using it.
MANUAL	MANUAL enables the RESET button and boots the system in manual mode.
REMOTE	REMOTE is used during initial system software installation. Once you have installed the software, you use this keyswitch position to troubleshoot system problems.
NORMAL	NORMAL disables the RESET button and boots the system in NORMAL mode.

Because the MANUAL keyswitch position enables the RESET button, which could be pressed accidentally during system operation, you should generally 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 is turned on and the internal power supply is operating properly.

You can use removable media (QIC tape or half-inch tape) to load system software. This media provides backup storage for the BTOS file system. If your base enclosure does not have a QIC tape drive slot, the removable medium for your system is half–inch tape. Refer to section 3 for information about using QIC and half–inch tape.

Powering Up the XE520

Caution: Whenever you power up the XE520, make sure that the removable medium drive is empty and the keyswitch is set to STOP.

To power up the XE520 system, complete the following:

1 Locate the **POWER ON/OFF** switch (refer to figure 2–2) at the rear of each enclosure in the system and turn each enclosure to the ON position.

When powering up enclosures in a multi–enclosure system, start with the base enclosure, then power up the second enclosure, the third enclosure, and so on.

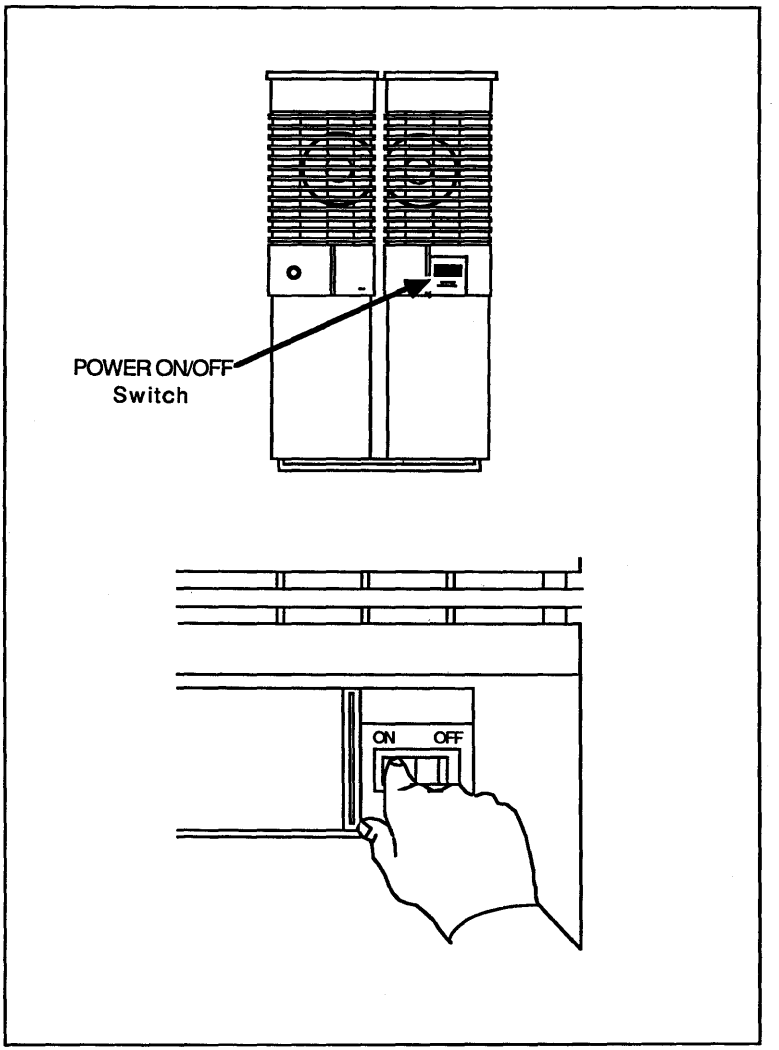
If you have powered up correctly, the power indicator at the front panel of each enclosure should be lit, the STATUS display at the base enclosure should show the value 00, and you should hear the fans and feel a steady flow of air from the rear ventilation grill of each enclosure.

If any of these three conditions does not occur after powering up the XE520, turn the XE520 off and contact your Unisys field engineering representative.

2 Once the enclosures are powered up, turn 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 *XE500 BTOS Administration Guide* for more information on operating modes.

Figure 2–2 **XE520 Enclosure POWER ON/OFF Switch**



Booting a BTOS Workstation from the XE520

You can operate a BTOS workstation as a standalone computer or as a cluster workstation connected to a master.

When you power up your workstation, the system first attempts to boot from a diskette in the floppy drive. If the floppy drive is empty or contains an unbootable diskette, the system then attempts to boot from the local hard disk. If the hard disk does not contain bootable software, the system then attempts to boot from the cluster master (the XE520, for example).

How you boot up your workstation determines the version of workstation BTOS that is loaded into the workstation. The version of workstation BTOS software that is loaded into your workstation determines whether the workstation operates in standalone or cluster mode.

BTOS workstations connected to the XE520 run as cluster workstations. As a result, the XE520 serves as the master, providing services that can be shared by the cluster workstations connected to it.

To configure your workstation to run in cluster mode, you must follow a separate bootup procedure for each of the following workstation configurations:

- a BTOS workstation on which system software has been installed (it contains a [sys]<sys>SysImage.sys file)
- a BTOS workstation with no standalone BTOS in its local file system

Booting a Workstation Having Standalone BTOS Software

Your workstation may have standalone workstation BTOS software stored on a local hard disk or floppy disk. When the workstation is powered up or rebooted, it loads that operating system into its central processing unit's (CPU's) onboard memory.

However, to access the XE520's storage devices and its local hard disk, you must configure your BTOS workstation as a cluster workstation. This downloads (sends) the workstation cluster–mode version of BTOS from the XE520 to your workstation.

One advantage to using your workstation in a standalone environment is that your operating system still runs if the XE520 is shut down. However, when the XE520 is brought up again, you must reboot your workstation (using the following procedures) to return it to the cluster mode.

Caution: If a workstation connected to the XE520 is attempting to operate as a master (that is, if a workstation with standalone workstation software is booted up locally while the XE520 is operating), the system will crash.

To boot your BTOS workstation from the XE520, complete the following:

1. Be sure that the XE520 has been successfully powered on (a 20 appears in the STATUS display).
2. While holding down the **Spacebar** on your BTOS workstation, power up or reboot your workstation.

   The system displays the Read Only Memory (ROM) number and then a series of characters (such as B, C, D, L, M, P, T:). The cursor appears after the colon.
3. Release the **Spacebar**.
4. Type the letter **T** (the T stands for type of operating system).

   The system displays **OS:** and the cursor. (The recommended operating system (OS) number may also be displayed.)

   This prompt is a request for the operating system number of the workstation. To learn the appropriate number for your workstation type, refer to Determining Your Workstation Operating System Number, later in this section.
5. Enter your operating system number after the **OS:** prompt.

6 Press **RETURN**.

The system responds by again displaying the series of characters, followed by the cursor.

7 Enter a **B** (the B stands for boot).

The system responds by displaying **L.....** This response indicates that the workstation is going through the boot–up process.

When the boot–up process is complete, the signon form appears on the screen. You can now use your workstation as part of the XE520 system.

Booting a Workstation Having No Standalone BTOS Software

If your BTOS workstation does not have a standalone version of BTOS stored locally on a hard disk or floppy disk, use the following procedure to boot it from the XE520:

1 Be sure that the XE520 has been successfully powered on (a 20 appears in the STATUS display).

2 Power up your BTOS workstation.

The workstation automatically boots from the XE520 when it is powered up or rebooted.

The system responds by displaying **L.....** This response indicates that the workstation is going through the boot–up process.

When the boot–up process is complete, the signon form appears on the screen. You can now use your workstation as part of the XE520 system.

Determining Your Workstation Operating System Number

When a BTOS workstation is booted from the XE520, the XE520 loads the workstation's operating system (or system image) into the workstation's local memory.

BTOS has been modified to make it compatible with different workstation hardware configurations. Therefore, there are unique BTOS versions for the different workstation models and configurations.

To determine your workstation model, locate the Unisys identification plate on the bottom of the workstation's CPU module.

Table 2–2 lists the workstations and their corresponding BTOS operating system numbers.

Table 2–2 **BTOS Workstation Operating System Numbers**

Workstation Model	Operating System Number
B24 (no local file storage)	200
B26 (hard disk)	250
B26 (floppy drives/no hard disk)	251
B26 (no local file storage)	252
B27 (hard disk)	125
B27 (floppy drives/no hard disk)	126
B27 (no local file storage)	127
B28 (hard disk)	240
B28 (floppy drives/no hard disk)	241
B28 (no local file storage)	242
B38 (hard disk)	230
B38 (floppy drives/no hard disk)	231
B38 (no local file storage)	232

Handling BTOS Workstation Bootup Problems

As part of the bootup process, a BTOS workstation runs through a self-diagnostic test. If an error message appears on the screen, refer to your Status Codes documentation.

If you suspect that you responded to a system prompt with incorrect information, reboot your workstation and complete the rest of the bootup procedure.

If you reboot your workstation and still receive unexpected system information or prompts, verify the following:

- Be sure that you are using the proper bootup procedure.
- For a workstation with a local standalone operating system, be sure that you correctly determined the workstation model/type and the correct operating system number for that type.

Powering Down the XE520

To power down the XE520, use the following procedure:

1. Make sure that all users are logged off, and that workstations and peripheral devices (printers, tape drives, etc.) are turned off.
2. Make sure that no tape is currently mounted.
3. Turn the keyswitch on 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 disappear, and the fans should stop operating.

Using QIC and Half–Inch Tapes

QIC tapes provide you with portable media on which to store information. On the XE520 system, they also provide you with the source for the initial system software and for future software updates. You can also use them as backup storage devices.

Handling QIC Tapes

When handling QIC tapes, remember the following:

- 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, you should pack (rewind) the QIC tape to ensure that the tape operates properly during I/O operations. If any of the following conditions apply, use the mQIC RETENSION command to rewind the QIC tape before using it:

- 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 cartridge storage (more than two weeks)
- physical shock to the cartridge (such as dropping it or dropping something on it)

For information on executing the mQIC RETENSION command, refer to Retensioning a QIC Tape, later in this section.

Figure 3–1 shows the components of a QIC tape. Figure 3–2 shows the positions of the write–protect plug for enabling and disabling write operations to the tape.

Figure 3–1 **Components of a QIC Tape**

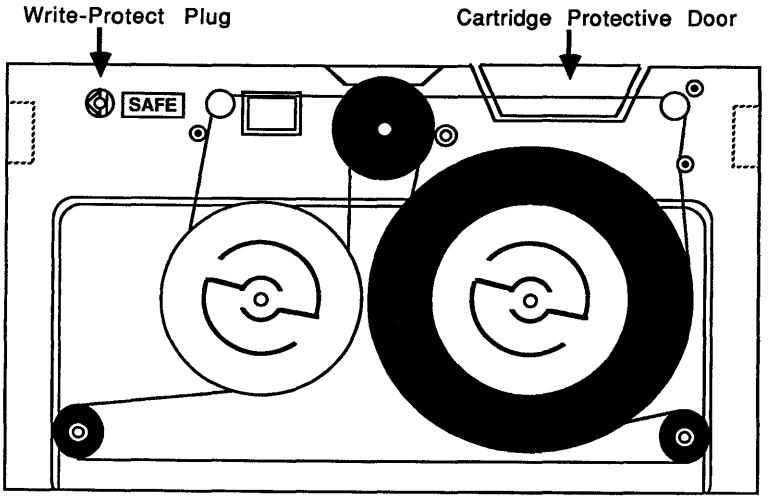
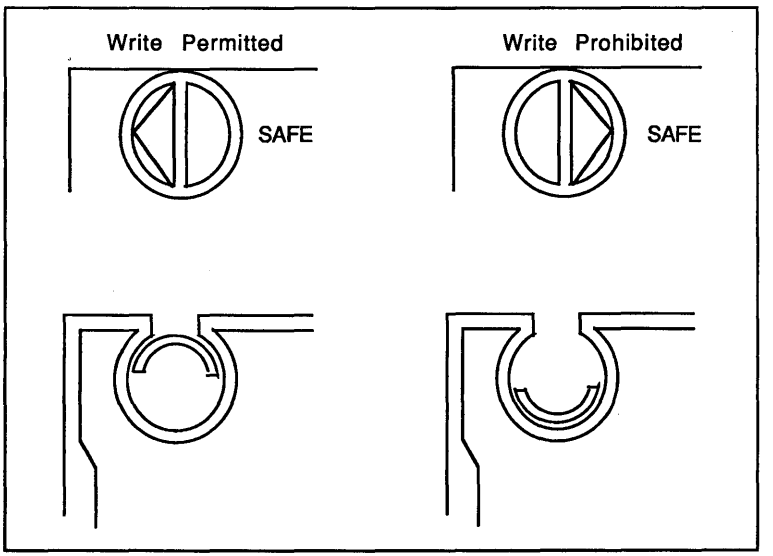


Figure 3–2 **QIC Tape Write–Protect Plug Positions**



Operating the QIC Tape Drive

You control the QIC tape drive using the front slide lever and the cartridge slot (shown in figure 3–3).

The front slide lever locks the QIC tape in the drive. The red indicator light on the right side of the drive is on whenever the system is retrieving or storing information.

Caution: Do not push the front slide lever when the indicator light is on; you could interrupt a write operation to the tape.

Inserting a QIC Tape

To insert a QIC tape, use the following procedure:

1. Be sure that the XE520 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 (refer to figure 3–4).
4. Push the QIC tape into the drive until it descends into the tape drive.
5. 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.

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

Figure 3–3 **The QIC Tape Drive**

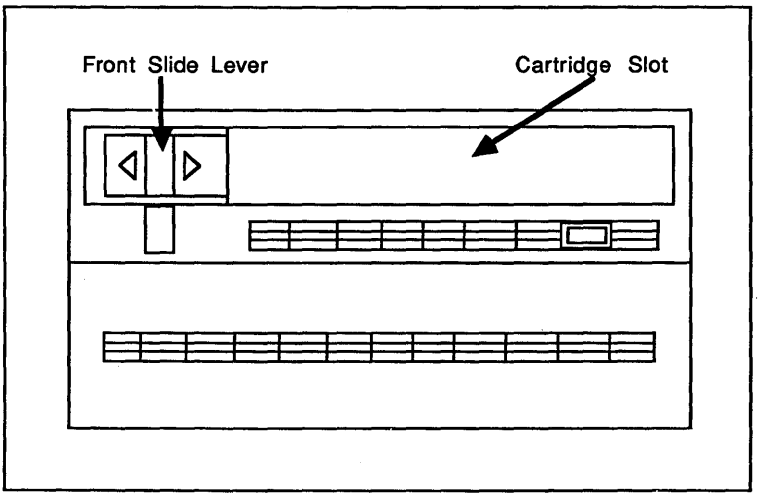
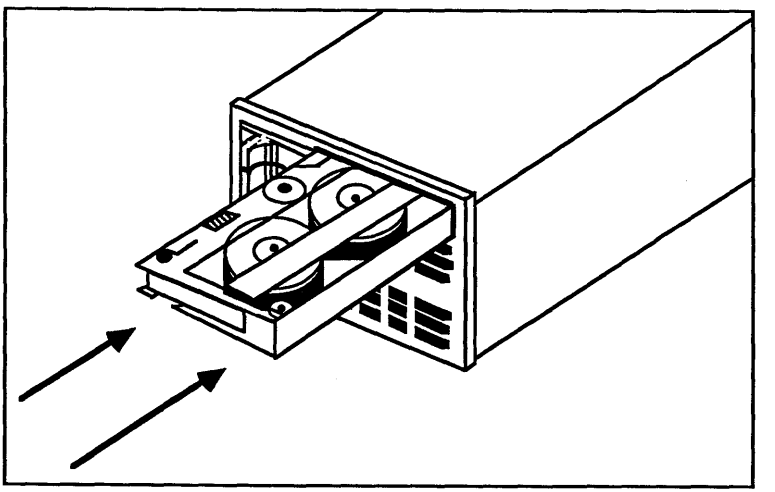


Figure 3–4 **Inserting the QIC Tape**



Removing a QIC Tape

To remove a QIC tape, use the following procedure:

1 Make sure the drive indicator light is off.
2 Move the front slide lever to the left until it reaches the lever stop.

   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.
3 Remove the QIC tape and return it to its case.

Retensioning a QIC Tape

You use the mQIC RETENSION command to rewind a QIC tape; this retensions (packs) the tape.

To retension a QIC tape using the mQIC RETENSION command, complete the following:

1 Insert the QIC tape into the QIC tape drive.
2 Type **mQic Retension** at your workstation's Executive command line.
3 Press **GO**.

Handling a Half–Inch Tape

When handling half–inch tapes, be careful not to touch the first few inches of the tape so that you do not lose the identifying information in the first tape file.

If either the tape or the header records are unreadable, the system scans the tape until it finds a valid data block. The TAPE RESTORE command recognizes the tape as a valid backup tape if the data block conforms to the format of a tape backup data block. However, if you attempt to append data to a tape that has unreadable tape or file headers, the tape backup commands return an error.

Caring for the Half-Inch Tape Drive

The tape transport (drive) has two heads, one behind the other, that float very close to the tape media, and read and write to it.

When the drive is writing, the first head writes, while the second head, set at low gain, reads behind it to verify the quality of the writing. When the drive is reading, the read head is set at high gain for more sensitivity.

If the drive is writing and the second head cannot read what the first head has just written, the system reports a write error.

If you are experiencing several errors, you should check to make sure that the drive heads are clean; dirty heads are the most frequent cause of errors. If you are using the drive heavily, you should clean the drive heads everyday. Otherwise, weekly cleanings are recommended.

Loading a Half-Inch Tape

To load a half-inch tape, complete the following:

- Attach a write-enable ring on the back of the tape reel of a 2400-foot, half-inch magnetic tape.
- Insert the reel into the tape drive and close the door.
- Press **Load/Rewind** (on a Cipher drive) or **Load/Online** (on a Pertec drive).
- The indicator light flashes while the tape is automatically loaded. The tape is finished loading when the light stops blinking and remains lit.
- Press **Online** (on a Cipher drive) or **Load/Online** (on a Pertec drive).

  From the point, the drive responds to commands from the system rather than from its control panel. If you have trouble loading the tape, refer to your tape drive's documentation.

Removing a Half–Inch Tape

To remove a half–inch tape, complete the following:

- Press **Online** (on a Cipher drive) or **Load/Online** (on a Pertec drive) to make sure that the tape drive is offline.
- Press **Unload** twice (on a Cipher drive) or **Rewind** twice (on a Pertec drive).

Installing the XE520 BTOS System Software

These software installation procedures assume that you are familiar with using portable media (QIC tapes and half–inch tapes). If you are unfamiliar with portable media, refer to section 3.

The workstation you use during installation must have a floppy disk drive and must be connected to the first Cluster Processor (CP00) in the XE520.

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

These workstation and tape drive restrictions are necessary because the system recognizes only the first processor of each processor type during certain parts of the installation and implementation procedures.

Updating versus Initializing

Initializing your system erases the existing data and files on the volume. Before you load new software, initializing your system ensures that all indicators and constants are set to prescribed conditions and values.

Updating your system allows you to save customized system configuration files. However, when you update your system, old and incompatible files that you do not delete may corrupt your system or unnecessarily use disk space. In addition, customized BTOS processor operating systems cannot be used and are overwritten during the installation procedure.

Note: Although it is not required, it is highly recommended that you initialize your system before you install the XEBTOS 7.0 system software.

If you choose to update your system rather than reinitialize it, you should be aware that your system may contain files that are incompatible with the XEBTOS 7.0 software.

When updating your system, you can save the following types of system configuration files:

- processor initialization files
- processor configuration files
- device configuration files (that is, files created through the mCREATE CONFIGURATION FILE utility)
- the queue index file, [sys]<sys>queue.index
- the printer spooler configuration files--for example, [sys]<spl>splcnfg.sys (refer to section 12)

Changes are required for some processor configuration and initialization files.

Initializing Your System

This section explains how to install the XE520 BTOS system software onto your system, and includes procedures for initializing your system disk.

The software covered in this section is available on QIC or half-inch tape. The installation procedure is nearly the same for each medium. Both media come with a floppy diskette that you insert into your workstation's floppy drive to start the installation procedures.

Hardware Requirements

Before installing XE520 BTOS system software, verify the following hardware requirements:

- The RS-422 cable must be attached to XE520 on channel 1a, 1b, 2a, or 2b on the CP board. The RS-422 cable must also be attached to the BTOS workstation in one of the cluster communication slots.
- The RS-232 cable (crossed pins) must be attached to the XE520 on either channel 3 of the CP00 or channel 10 of the TP00. The RS-232 cable must also be attached to the BTOS workstation in the RS-232 slot.

Installing the XE520 Software

To install the XE520 system software, complete the following:

1. Insert the appropriate floppy disk for your workstation into the disk drive.
2. Power up or reset the workstation.

   The workstation boots from the floppy disk and then displays the Signon screen.
3. At the Signon screen, press **GO**.
4. Verify that the XE520 is powered up and that the key on the front status panel is in the STOP position.
5. Depending on the medium you use, load either the QIC tape or the half-inch tape.
6. Turn the front panel key on the XE520 to the MANUAL position.

   The XE520 now boots up from the tape. This takes approximately five minutes.

   When it finishes, the system displays a menu containing the following options:

   0 = Set Path / Enter Password
   1 = Initialize a Volume
   2 = Restore Volume from Tape
   3 = Change Volume Name
   4 = Backup a Volume to Tape
7. Type **1**.
8. Press **RETURN**.

   The system displays the mIVOLUME command form parameters as a series of prompts asking you to supply information.

   Refer to figure 4–1 for the recommended responses to these mIVOLUME prompts for XE520–4 systems. Refer to figure 4–2 for the recommended responses to these mIVOLUME prompts for XE520–5 and –6 systems.

Figure 4–1 **Recommended mIVOLUME Responses for XE520–4 Systems**

Device Name: :	**d1**
Device Password: :	**d1**
Volume Name: :	(Enter a volume name)
Volume Password: :	
System Image sectors: [512] :	
Log file sectors: [32] :	
Crash file size in sectors: [1536] :	
Maximum number of directories on volume: [0] :	**300**
Maximum number of files on volume: [0] :	**4000**
Primary file headers only? [No] :	
Maximum number of files in Sys directory: [0] :	**400**
Sys directory password: :	
Write protect Sys directory? [No] :	
Suppress format of medium? [No] :	
Surface tests: [9] :	(Minimum of) **3**
Debug? [No] :	
Bad spots: :	
Bad spots: :	
Bad spots: :	

Figure 4–2 **Recommended mIVOLUME Responses for XE520–5 and –6 Systems**

Device Name: :	**s0**
Device Password: :	**s0**
Volume Name: :	(Enter a volume name)
Volume Password: :	
System Image sectors: [512] :	
Log file sectors: [32] :	
Crash file size in sectors: [1536] :	
Maximum number of directories on volume: [0] :	**600**
Maximum number of files on volume: [0] :	**8000**
Primary file headers only? [No] :	
Maximum number of files in Sys directory: [0] :	**800**
Sys directory password: :	
Write protect Sys directory? [No] :	
Suppress format of medium? [No] :	
Surface tests: [9] :	(Minimum of) **3**
Debug? [No] :	
Bad spots: :	
Bad spots: :	
Bad spots: :	

9 Respond to each prompt, pressing **RETURN** after each response.

Note: If you need to change a response after you have pressed **RETURN**, continue to respond to the remaining prompts, pressing **RETURN** after each. After responding to the final prompt, continue to press **RETURN** to cycle through the prompts until you reach the one you want to change.

10 After responding to the final prompt, press **GO**.

If your disk is unformatted or corrupted, the system prompts you with the following message:

*******NO Valid Initial Home Block.**
Please enter the device type of the disk:

If you see this prompt, refer to Formatting a New or Corrupted Disk, later in this section

If your disk is not unformatted or corrupted, the system displays the following message:

Do you wish to ERASE this volume?
(Press Y to confirm, N to deny, or Q to quit the utility)

11 Press **Y**.

The system begins formatting the disk. This can take 15 minutes or more, depending on the size of the disk and the number of surface tests you specified. When it finishes, the system displays the prompt:

Please insert system tape, then hit <CR> key to reload menu program.

12 Press **RETURN**.

The menu returns to the screen.

13 Press **0**.

14 Press **RETURN**.

The system prompts you for the name that you assigned to your volume in step 8.

15 Enter your volume name.

16 Press **RETURN**.

The system prompts you for the password, if any, that you assigned to your volume in step 8.

17 Press **RETURN**.

The system prompts you for a directory name.

18 Enter **Sys**.

19 Press **GO**.

The menu returns to the screen.

20 Press **2**.

21 Press **RETURN**.

The system displays a series of prompts asking you to supply information.

22 Enter the following in response to the system prompts, pressing **RETURN** after each prompt:

Archive file: **[QIC]8** or **[Tape]8**
File list from: <*>*
To file list: <*>*
Overwrite ok [Prompt]: **Yes**
Confirm each [No]:
Sequence number [1]:
Merge with existing file [No]:
List file only [No]:
Log file:

23 Press **GO**.

The system displays the following prompt:

Configuration file cannot be found. Please enter following information:

<GO>/<ESC> to exit interaction, <CR> to continue.
Record size [10240]:

24 Press **RETURN**.

The system displays the following prompt:

Rewind on completion [yes]:

25 Press **RETURN**.

The system displays the following prompt:

High speed tape [yes]:

26 Press **GO**.

The system restores the files. This procedure takes approximately ten minutes. When it finishes, the system displays the following prompt:

Please insert system tape, then hit <CR> key to reload menu program.

27 Remove the tape from the drive.

28 Turn the key on the front panel of the XE520 to STOP then to REMOTE.

The XE520 boots from its own disk.

29 Remove the floppy from your workstation drive.

This completes the initial software installation. You are now ready to proceed with the configuration of your system.

Formatting a New or Corrupted Disk

When the XE520 boots up from the tape at the beginning of the software installation procedures, the system displays a menu containing the following options:

0 = Set Path / Enter Password
1 = Initialize a New Volume
2 = Restore Volume from Tape
3 = Change Volume Name
4 = Backup a Volume to Tape

If you choose option 1 and your disk is unformatted or corrupted, the system prompts you with the following message:

*******NO Valid Initial Home Block.**
Please enter the device type of the disk:

In response to this prompt, complete the following:

1 Type **new**.

2 Press **RETURN**.

The system displays the following series of prompts:

Removable Media?:
Cylinders Per Disk:
Tracks Per Cylinder:
Sectors Per Track:
Step Rate Code:

3 Respond to each prompt according to the type of drive on your system.

Refer to table 4–1 for a list of the drive types and the appropriate prompt responses for each.

4 Press **RETURN** after completing the information for each prompt.

5 After responding to the final prompt, press **GO**.

The system begins formatting the disk. This can take 15 minutes of more, depending on the size of the disk and the number of surface tests you specified. When it finishes, the system displays the prompt:

Please insert system tape, then hit <CR> key to reload menu program.

6 Continue with step 12 in the software installation procedures.

Table 4–1 **XE520 Drive Types and Menu Responses**

Drive Type	Menu Responses
Atasi 46 Mb Drive	Removable Media?: **No** Cylinders Per Disk: **645** Tracks Per Cylinder: **7** Sectors Per Track: **16** Step Rate Code: **0**
Ball 100 Mb Drive	Removable Media?: **Yes** Cylinders Per Disk: **1024** Tracks Per Cylinder: **5** Sectors Per Track: **32** Step Rate Code:
CDC 300 Mb Drive	Removable Media?: **Yes** Cylinders Per Disk: **823** Tracks Per Cylinder: **19** Sectors Per Track: **32** Step Rate Code:
CDC 340 Mb Drive	Removable Media?: **No** Cylinders Per Disk: **711** Tracks Per Cylinder: **24** Sectors Per Track: **32** Step Rate Code:
CDC 675 Mb Drive	Removable Media?: **No** Cylinders Per Disk: **840** Tracks Per Cylinder: **40** Sectors Per Track: **32** Step Rate Code:

Table 4–1 **XE520 Drive Types and Menu Responses** (continued)

Drive Type	Menu Responses
Fujitsu 80 Mb Drive	Removable Media?: **No** Cylinders Per Disk: **589** Tracks Per Cylinder: **7** Sectors Per Track: **32** Step Rate Code:
Hitachi 51 Mb Drive	Removable Media?: **No** Cylinders Per Disk: **699** Tracks Per Cylinder: **7** Sectors Per Track: **16** Step Rate Code: **0**
Hitachi 85 Mb Drive	Removable Media?: **No** Cylinders Per Disk: **823** Tracks Per Cylinder: **10** Sectors Per Track: **16** Step Rate Code: **0**
Maxtor 53 Mb Drive	Removable Media?: **No** Cylinders Per Disk: **903** Tracks Per Cylinder: **5** Sectors Per Track: **16** Step Rate Code: **0**
Maxtor 143 Mb Drive	Removable Media?: **No** Cylinders Per Disk: **918** Tracks Per Cylinder: **15** Sectors Per Track: **16** Step Rate Code: **0**
Memorex 166 Mb Drive	Removable Media?: **No** Cylinders Per Disk: **823** Tracks Per Cylinder: **10** Sectors Per Track: **32** Step Rate Code:
Micropolis 52 Mb Drive	Removable Media?: **No** Cylinders Per Disk: **753** Tracks Per Cylinder: **6** Sectors Per Track: **16** Step Rate Code: **0**

Table 4–1 **XE520 Drive Types and Menu Responses** (continued)

Drive Type	Menu Responses
Micropolis 85 Mb Drive	Removable Media?: **No** Cylinders Per Disk: **1024** Tracks Per Cylinder: **8** Sectors Per Track: **17** Step Rate Code: **0**
Toshiba 85 Mb Drive	Removable Media?: **No** Cylinders Per Disk: **830** Tracks Per Cylinder: **10** Sectors Per Track: **17** Step Rate Code: **0**
Unknown Drive	Removable Media?: **No** Cylinders Per Disk: **300** Tracks Per Cylinder: **7** Sectors Per Track: **16** Step Rate Code: **0**

Section 5

Installing Workstation Software

To install BTOS workstation software onto the XE520 system disk, you use system utilities that you run from your workstation.

Before you can use your workstation for installation procedure, it must:

- have a floppy disk drive
- be connected to the first Cluster Processor (CP00) in the XE520

The installation diskettes are divided into the following packages:

- B*xx*ST1 (where *xx* is 26, 27, 28, or 38 depending on your workstation model), which contains the BTOS standard software.
- B25SU1, which contains workstation utilities.
- B25CL1, which contains the cluster workstation operating systems.
- B25LD1, which contains the workstation language development software.

Each workstation software release diskette is appropriately marked. As you proceed through the installation procedure, the installation utility tells you which diskette to insert.

Note: Before the XE520 can support workstations, you must install the following required packages: B*xx*ST1, B25SU1, and B25CL1. Although the other packages are optional, software contained in them may be required by some BTOS environmental software.

The procedures for installing standard workstation software and the workstation utilities apply to the BTOS II 1.1 release of BTOS workstation system software.

Installing Standard Workstation Software

The standard workstation software allows you to boot the workstation from the XE520 and to install the BTOS workstation utilities.

To install the standard workstation software, complete the following:

1. Power on your XE520.
2. Power on your BTOS workstation.

   Remember that the workstation must have a floppy drive and must be connected to CP00 of the XE520.
3. Insert the first B*xx*ST1 diskette into [f0].

   The B*xx* designation should match the workstation you are using. For example, if you are using a B38 workstation, you would use diskette B38ST1–1.
4. Reset your workstation.

   The workstation boots from the floppy and the system displays the Signon screen.
5. Remove the write–protect tab from the second B*xx*ST1 diskette.
6. Remove the first B*xx*ST1 diskette and insert the second B*xx*ST1 diskette.
7. Type **XE520** in the **User name** field.
8. Press **GO**.

   The system displays a prompt about the old software in the <B20> directory.
9. Press one of the following:
   - **GO** to continue installation

     If you press **GO**, the system displays the PATH command form and prompts you to enter a password if one exists. Proceed to step 10.
   - **ACTION–FINISH** to abort the installation

     If you press **ACTION–FINISH**, the system stops the installation and displays another BTOS Executive command line. Before attempting to install the software, make a backup copy of your <B20> directory.
10. Enter a password, if applicable.

11 Press **GO**.

The system copies other files onto the XE520 system.

When these operations are complete, the systems asks you if you want to overwrite the [!sys]<sys>Sys.cmds file (if it exists).

12 Choose one of the following:

- If you are installing this workstation software as part of an initial XE520 software installation, press **GO** to overwrite the [!sys]<sys>Sys.cmds file.

  The system executes various commands and prompts you to reinsert the first B*xx*ST1 diskette. Proceed to step 13.

- If you are updating your workstation software with this workstation release, you should not overwrite your [!sys]<sys>Sys.cmds file. Press **CANCEL** and then **GO** to prevent the current [!sys]<sys>Sys.cmds file from being overwritten.

  The system executes various commands and prompts you to reinsert the first B*xx*ST1 diskette. Proceed to step 13.

  Note: If you chose to overwrite your [!sys]<sys>Sys.cmds file, or do so accidentally, you must add the master commands to this file after you have installed the workstation software (refer to your Standard Software documentation).

13 Remove the second B*xx*ST1 diskette and reinsert the first B*xx*ST1 diskette.

14 Press **GO**.

The system executes various commands.

When the software installation is complete, the system displays instructions for rebooting the workstation from the XE520 (refer to figure 5–1), and prompts you to press **GO** to continue.

15 Press **GO**.

The system displays instructions for continuing the software installation.

16 Remove the first B*xx*ST1 diskette.

17 Replace the write–protect tab on the second B*xx*ST1 diskette and store the diskettes in a safe place.

18 Reboot your workstation according to the instruction in figure 5–1.

Figure 5–1 **Rebooting Instructions**

REBOOTING THE WORKSTATION

Note the reboot instructions below:

A. Hold down the space bar and simultaneously reset the system.
B. When B,C,D,L,M,P,T: appears on the screen, enter T.
C. The video will then display OS:.
   1. If this system has a winchester disk, enter 240 then press return.
   2. If this system has only floppy disk storage, enter 241 then press return.
D. The video will again display B,C,D,L,M,P,T:. Enter B and the system will complete the boot process.

Installing the Workstation Utilities Software

The workstation utility software (contained on six diskettes) installs workstation utilities and creates their associated command forms.

During the installation procedure, you will have the option of installing certain utilities.

To install the workstation utilities software, complete the following:

1 Sign on to your workstation system.
2 Insert the first B25SU1 diskette into [f0].
3 Type **XESoftware Installation** at the BTOS Executive command line.
4 Press **GO**.

   The system copies various files and then asks you if you want to overwrite the [!sys]<sys>Sys.cmds file (if it exists).

5 Choose one of the following:

- If you are installing this workstation software as part of an initial XE520 software installation, press **GO** to overwrite the [!sys]<sys>Sys.cmds file.

  The system executes various commands and prompts you to insert the second B25SU1 diskette. Proceed to step 6.

- If you are updating your workstation software with this workstation release, you should not overwrite your [!sys]<sys>Sys.cmds file. Press **CANCEL** and then **GO** to prevent the current [!sys]<sys>Sys.cmds file from being overwritten.

  The system prompt you to insert the second B25SU1 diskette. Proceed to step 6.

  Note: If you chose to overwrite your [!sys]<sys>Sys.cmds file, or do so accidentally, you must add the master commands to this file after you have installed the workstation software (refer to your Standard Software documentation).

6 Remove the first B25SU1 diskette and insert the second B25SU1 diskette.

7 Press **GO**.

The system begins installing various utilities and then prompts you to install the first optional utility, BACKUP VOLUME.

Note: For a description of the utilities, refer to your Standard Software documentation.

8 To install the BACKUP VOLUME utility and the other optional utilities on the remainder of the B25SU1 diskettes, press **GO** in response to the utility's prompt.

If you decide not to install a particular optional utility, press **CANCEL** and **GO** in response to the utility's prompt.

As you cycle through the optional utility prompts for a given B25SU1 diskette, the system prompts you to remove the diskette and insert the next one.

When you have cycled through the utilities on the last B25SU1 diskette, the system tells you that the workstation utility installation is complete.

9 Remove the last B25SU1 diskette and store the diskettes in a safe place.

10 Reboot the XE520 and all the workstations in the cluster.

Installing Cluster Workstation Software

The cluster workstation software (contained on five diskettes) allows workstations to run within a BTOS cluster environment.

To install the BTOS cluster workstation software, complete the following:

1 Sign on to your workstation.

2 Insert the first B25CL1 diskette into [f0].

3 Enter **XESoftware Installation** at the BTOS Executive command line.

4 Press **GO**.

The system prompts you to power down all other workstations.

5 After you have verified that all other workstations in the cluster are powered down, press **GO**.

The system automatically installs the following utilities:

Cluster Status
Disable Cluster
Resume Cluster

The system then prompts you to have a list of the model numbers of the workstations on your system.

6 Press **GO**.

The system displays a prompt asking you if you have B24 workstations in your cluster.

7 Press **GO** if you have B24 workstations in your cluster, or **CANCEL–GO** if you do not.

8 When the system prompts you, remove the first B25CL1 diskette and insert the second B25CL1 diskette.

9 Press **GO**.

During the remainder of the cluster software installation, you are prompted in sequence to install the various cluster workstation operating systems.

10 If your workstation is of the type displayed in the prompt, press **GO**. If it is not, press **CANCEL–GO**.

   As you progress through the cluster operating system prompts for a given B25CL1 diskette, the system prompts you to remove the diskette and insert the next one.

   When you have cycled through the cluster operating systems on the last B25CL1 diskette, the system tells you that the cluster software installation is complete.

11 Remove the last B25CL1 diskette and store the diskettes in a safe place.

12 Reboot all the workstations in the cluster.

Installing Workstation Language Development Software

The Language Development software (contained on two diskettes) consists of software tools that support BTOS programming languages.

To install the workstation Language Development software, complete the following:

1 Sign on to your workstation.

2 Insert the first B25LD1 diskette into [f0].

3 Type **XESoftware Installation** at the BTOS Executive command line.

4 Press **GO**.

   The system prompts you to power down all other workstations in the cluster.

5 After powering down the other workstations in the cluster, press **GO**.

   The system begins installing the Language Development software.

   The system displays a series of prompts that allow you to decide which Language Development tools you want to install.

6 To install a tool, press **GO** in response to the prompt.

   If you decide not to install a particular tool, press **CANCEL–GO** in response to the prompt.

   As you cycle through the prompts, the system prompts you to remove the diskette and insert the next one.

When you have cycled through the tools on the second B25LD1 diskette, the system tells you that the Language Development software installation is complete.

7 Remove the second B25LD1 diskette and store the diskettes in a safe place.

8 Reboot all the workstations in the cluster.

Section 6

Overview of System Configuration Files

XE520 BTOS uses several text files to define the system's hardware and software configurations. The system reads these text files when BTOS is loaded into the XE520 processor boards (that is, when the system is booted).

Note: The system reads different sets of configuration files depending on the positions to which the keyswitch can be turned to boot the system. Refer to appendix B for details.

The following are the types of configuration files:

- Master configuration file

  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. It contains an entry for each processor in the system, except the master processor. The entries define the version of XE520 BTOS it loads on the processors. Other entries in this file define the processor configuration files for applicable processors.

- Processor configuration file

  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 contains entries that define the dynamic block allocation parameters for the processor. These parameters determine the size of I/O communications buffer space.

- Processor initialization file

  A processor initialization file defines the system services that run on a processor once that processor has booted up. Entries in this file are normally the run files of the desired system services.

- Queue index file

  The queue index file defines the queues for which the Queue Manager is responsible.

- Spooler configuration file

  The spooler configuration file contains entries for each printer in the system. A printer entry defines parameters such as where the printer is connected and what print queue the printer is to serve.

- I/O device configuration files

  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. You can modify the printer and tape drive configuration files and create other I/O device configuration files by using the mCREATE CONFIGURATION FILE command.

This section describes the configuration files related to the XE520 processor boards, disk drives, and I/O device configuration files. The processor initialization files, queue index file, and spooler configuration file are described in section 7.

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 for the other processor boards and the order in which to load them.

When you boot the system in restricted mode, the master configuration file [sys]<sys>Master.r.cnf is used. This master configuration file allows enough of the system to be brought up so that you can run the mBTOS CONFIG utility.

The master configuration file contains the following line entries:

- the nowatchdog entry

  The line **nowatchdog** must be included in the master configuration file.
- the watchdog entry
- processor BTOS entries

  BTOS entries following the nowatchdog entry for each processor in the XE520 system, except the master processor. Each of these entries determines the version of BTOS to be run on the processor. Entries are ordered according to how their corresponding processors are configured in the XE520 enclosures. The first entry corresponds to the first processor board after the master processor, the next entry corresponds to the next processor board, and so on.
- include statement entries for processor configuration files

  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.

Caution: You should not include the master processor and SP configuration files 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>Spnn.cnf, where nn is the SP number.

By convention, processor configuration file names always have the suffix ".cnf."

Sample Master Configuration File

The following is a sample master configuration file:

```
NoWatchDog
Cp          [sys]<sys>CpBtos.sys
Include     [sys]<sys>Cp00.cnf
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, 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>ZpBtos.sys, where Zp 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>Zpnn.cnf, where Zp represents the processor type and nn is the processor's number within the system. This number uniquely identifies processor boards of the same type.

Note: The master configuration file must contain 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. For CPs and TPs, they 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>Zpnn.cnf, where Zp is the processor type (such as FP, CP) and nn is the processor number (such as 00, 01, or 02). For example, the first CP is CP00, the second CP is CP01, the third CP is CP02, and so on.

Refer to appendix A for a listing of default processor configuration files.

Dynamic Block Allocation

The system uses X, Y, and Z blocks to provide buffer space for various types of communications.

The system uses X blocks exclusively in the CP for data transmission to and from the workstations. The number of X blocks affects both the ability of the workstations to boot and workstation 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. The default size of an Y block is 2560 bytes.

Z blocks are the small IPC communications buffers, and the system uses them for the same purpose as Y blocks. 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.

You allocate X, Y, and Z blocks by adding a line (or lines) in the processor configuration file. These line entries have the following form:

Blocktype=type, Number=nn [, Size=ss]

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 multiplied by 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. You can modify the parameters used by a BTOS operating system by customizing the operating system. Refer to your Customizer documentation for more information.

For X blocks, if the configuration file contains values for the number of X blocks or the size of the X blocks, then the system uses those values to allocate space. Otherwise, the space configured when the CP's operating system was generated is used.

Refer to appendix A for a listing of default processor configuration files.

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 connected to it. The FP or DP reads this file after its operating system has been loaded and it has booted up.

Two entries in this configuration file define the number of Y and Z blocks to be allocated. Each disk drive connected to the processor uses one of these two entries, which 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=23
BlockType=z, Number=95
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
DEBUGGER
```

The first two lines define the number of Y and Z blocks allocated (23 and 95, 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 through 5 are s0 through s5, respectively.

The following lines represent the entries of a default FP00 configuration file:

```
BlockType=y, Number=23
BlockType=z, Number=95
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 (23 and 95, respectively).

The next four define disk drive parameters. This FP supports four 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 both of the following:

- the parameters used when the FP or DP operating system was generated
- 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 your Customizer documentation.) This entry defines the disk number, device name, and device password for each disk controlled by the FP or DP.

You can modify disk information as defined in the processor's operating system with the information for that disk's entry in the FP's or DP's configuration file. However, if you delete the disk's entry in the configuration file, the system uses the information in the processor's operating system.

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, you deleted the entry for d4 from this file, the system uses the default information for d0. This would cause a duplicate device name, and the system would return Erc 236 (**Invalid device specification**) 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, and modems connect to RS–232–C ports. Workstation 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.

Caution: A CP configuration file must contain 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 to indicate optional parameters):

[async] ChannelNumber, [Parit=n], [Speed=n], [StopBits=n], [CharBits=n], [FlowGen], [FlowAct], [FlowAny], [XOFFlev=n], [XONLev=n], [modem], [connect=ctos]

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 XE520 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. However, you must still include 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]

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 307 k (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 307 k is used.

Default Cluster Processor Configuration File

The following lines represent the entries of a default CP configuration file:

```
BlockType=x, Number=16
BlockType=y, Number=6
BlockType=z, Number=60
async 3, speed=9600, parity=none, stopbits=1,
charbits=8, connect=ctos
line 1, clock=2M
line 2, clock=2M
Debugger
```

This CP is supporting one serial printer, connected to channel 3, and two workstation cluster lines. The workstation cluster lines are set to run at a baud rate of 2M.

Storage Processor Configuration File

The SP configuration file contains dynamic block allocation entries for Y and Z blocks. The following is a default SP configuration file:

BlockType=y, Number=4
BlockType=x, 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.

I/O Device Configuration Files

XE520 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 following form:

[sys]<sys>drivename.cnf

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 A.

Printer Configuration Files

Default configuration files for a parallel printer and a serial printer 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

You reference the system's printer configuration files in the printer spooler configuration file, [sys]<spl>SplCnfg.sys. Because you can modify the printer spooler configuration file, printer configuration files can have any file name. Also, you can use the same printer configuration file 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 found in the processor configuration file.

You can modify or create printer configuration files by using the mCREATE CONFIGURATION FILE utility (refer to section 11).

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

You can modify or create half–inch tape drive configuration files by using the mCREATE CONFIGURATION FILE utility (refer to section 11).

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 11).

Communication Device Configuration Files

You must create configuration files for communication devices, such as modems or terminals, using the mCREATE CONFIGURATION FILE utility (refer to section 11). Refer to the device's supporting documentation to determine parameter values for which the mCREATE CONFIGURATION FILE utility will prompt you.

Overview of System Services

XE520 BTOS provides system services that control access to software applications and peripheral devices. These system services run on XE520 processor boards.

The following system services are installed on the appropriate processor boards as part of the system software installation procedure:

- the AdminAgent
- tape servers (for half–inch tape drives)
- QIC server (for QIC tape drive)

The BTOS software uses processor initialization files to install the system services onto appropriate processor boards. The system reads these initialization files when BTOS is loaded into the XE520 processor boards (that is, when the system is booted).

This section describes the processor initialization and system services configuration files. Section 8 describes how you create and modify them to suit your system's configuration.

A general description of processor initialization files is provided first. The system services are described in 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 normally has an initialization file. Processor initialization file names take the form [sys]<sys>initZpnn.jcl, where Zpnn is the processor designation (for example, FP00, CP02).

The initialization file is a text file that contains 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 you can include in a JCL command file. The format for a run statement in a JCL file is $run runfile, parameters.

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. You use one or more run statements 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

You can divide a BTOS processor's onboard memory into regions called partitions. You can assign to a program being run by the processor 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 the processor runs several programs on the processor at the same time. Each program actually shares processing time on the processor.

There are two types of BTOS partitions:

- primary parition

  The system automatically sets up a primary partition 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 partition

  You must create secondary partitions 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 you want to install new system services after your system is delivered, you may want to create secondary partitions for them to enhance overall system efficiency. Refer to the *XE500 BTOS Administration Guide* for guidelines and procedures for installing system services.

AdminAgent

AdminAgent is the system service that manages the execution of master utilities on XE520 processors.

Note: To run an AdminAgent on a processor, the processor must have 768 Kb of memory.

The AdminAgent is run in the primary partition of the processor whose initialization file contains the AdminAgent run statement. The run file for AdminAgent is [sys]<admin>MfAdminAgent.run.

Only one processor can be running AdminAgent. Also, using this AdminAgent mechanism, only one user can be executing master utilities at a time.

If you want to run AdminAgent, include the run statement for [sys]<admin>MfAdminAgent.run in a processor's initialization file.

Note: Some XE520 commands require that the AdminAgent is running on the master (FP or DP) processor board. Refer to the *XE500 BTOS Operations Guide* for more information about XE520 commands.

Obtaining AdminAgent Status Information

To obtain status information about the AdminAgents in the system, use the mADMIN AGENT STATUS command.

For a AdminAgent, the status information indicates the processor on which the AdminAgent is running.

To execute the mADMIN AGENT STATUS command, enter **mAdmin Agent Status** at the BTOS Executive command line and press **GO**. Status information is displayed at the terminal screen.

Queue Manager

The Queue Manager controls the various queues in which the system stores related tasks 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.

For more information about the Queue Manager, refer to section 10.

Printer Spooler Managers

The printers connected to an XE520 processor board (either a CP or TP) are controlled by a printer spooler manager running on that processor. A printer spooler manager can also run locally at a workstation to control any printers connected to the workstation. Regardless of where spooler managers are run, all spooled printing queues in the system are controlled by the XE520 Queue Manager.

The 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 printer spooler can only control printers connected to serial channels 1, 2, and 3. You must install a printer spooler in each CP or TP to which printers are connected.

To load the printer spooler manager, include the following commands in the CP or TP processor initialization file:

$Run [sys]<sys>CreatePartition.run,*size*,*part*
$Run [sys]<admin>mInstallServer.run,*part*,&
[sys]<sys>mSpoolerMgr.run,*config*

The parameters for CreatePartition.run are as follows:

size	size is the size of the partition required by the printer spooler manager. The recommended size is 70 Kb.
part	part is the name of the partition the printer spooler manager is installed into. This name can be up to 12 characters long and must be used in the mINSTALL SERVER command.

The parameter for mSpoolerMgr.run is as follows:

config	config is the name of the Printer Device Configuration file. Refer to your Standard Software documentation for more information about the Printer Device Configuration file.

When a 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 a printer spooler manager run statement must be the last statement in any initialization file in which it appears. The system never executes a run statement in an initialization file that appears after such a printer spooler manager run statement. You can overcome this limitation by installing the 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, you should install the small printer spooler into the primary partition.

mQIC Server

During system software installation, the mQIC server is installed automatically on a processor board that is located in the primary enclosure. This primary enclosure contains the QIC Interface board. It is recommended that you place the mQIC Server on a lightly loaded board (usually an FP or TP).

To install mQIC Server, include the following commands in the FP or TP processor initialization file:

$Run [sys]<sys>CreatePartition.run,*size*,*part*
$Run [sys]<admin>mInstallServer.run,*part*,&
[sys]<admin>mQICServer.run,*buffers*,*length*,*slot*

The parameters for CreatePartition.run are as follows:

size — size is the size of the partition required by the mQIC Server. The recommended size is 100 Kb.

part — part is the name of the partition the mQIC server is installed into. This name can be up to 12 characters long and must be used in the mINSTALL SERVER command.

The parameters for mQICServer.run are as follows:

buffers — buffers is the maximum number of buffers to be allocated. The maximum value is 255 and the default is 128.

When installed, the mQIC Server allocates 512–byte buffers until it runs out of memory in its partition, or until it reaches the maximum number of buffers. If mQIC Server cannot allocate at least 40 buffers, it writes an error message to the system log and terminates.

This parameter is currently ignored. It exists as a placeholder so that old SysInit.jcl files continue to function without modification. The system attempts to allocate as many buffers as possible in the partition in which the mQIC Server is loaded. Refer to Buffering for the QIC and Half-Inch Tape Servers, later in this section.

length — length is the assumed cartridge length (in feet). The default is 600. This parameter is currently ignored. It exists as a placeholder so that old SysInit.jcl files continue to function without modification.

slot — slot is the hexadecimal number that identifies the slot containing the QIC board. The default is slot 77.

mTape Server

mTape Server is a system service that must be installed on the XE520 board or boards that are connected to a half-inch tape drive (DPs or SPs only).

Note: You should not run mTape Server on the same processor board as the mQIC Server.

To install the mTape Server, include the following commands in the processor initialization file for each processor board attached to a half-inch tape drive:

$Run [sys]<sys>CreatePartition.run,*size*,*part*
$Run [sys]<admin>mInstallServer.run,*part*,&
[sys]<admin>mTapeServer.run,*units*,*buffsize*

The parameters for CreatePartition.run are as follows:

size — size is the size of the partition required by the mTape Server. The recommended value is 180 Kb. To to determine the minimum size of the partition for mTapeServer, use the following formula:

(3 x buffsize) + 20K

part — part is the name of the partition the mTape server is installed into. This name can be up to 12 characters long and must be used in the mINSTALL SERVER command.

The parameters for mTapeServer.run are as follows:

units — units is the number of half-inch tape drives connected to this processor.

buffsize — buffsize is the size of the mTape Server buffers. The default is 10 Kb. This parameter must be between 1 and 64.

Buffering for the QIC and Half-Inch Tape Servers

It is very important to install the tape servers with correct buffering. Buffering determines the efficiency of disk/tape data transfers.

mQIC Server

With mQIC Server, you can enforce a limit by experimenting with partition sizes. Create a partition just large enough for mQIC Server to install, and just small enough for mQIC Server not to install if you decrease the size of this partition. That size is the size required for the server code and 40 buffers (20 Kb), which is the minimum partition size. Then, for every 1Kb the partition size is increased, two more buffers will be allocated.

mQIC Server, unlike mTape Server, always requires buffers, regardless of where the tape utility is running.

When writing to tape, the QIC drive's firmware tries to keep the tape moving in the following manner. If the buffer is written and there is not another buffer waiting to be written, the firmware advances the tape as far as it would take to write the buffer again. It does this in anticipation that there will be a buffer waiting to be written by the time it completes this advance of the tape. If there is no buffer waiting, the drive rewinds back to the end of the record.

mTape Server

As long as the record size is the same, a half–inch tape created by a server with plenty of buffers is physically identical to one created by a server with few buffers. mTape Server performance is not affected by how many times it starts and stops.

The two most important considerations are speed and memory. In general, the more memory used, the faster the speed. The minimum arguments to mTape Server are one for the number of tape drives (*units*) and one for the buffer size (*buffsize*). The actual space required for the buffers is as follows:

units x *buffsize* x 2

Note: mTape Server uses buffers only if the tape utility is running on a different board from mTape Server. Therefore, if you install the AdminAgent on the same board as mTape Server and use XE520 mCommands to access the server, you can specify one tape drive and 1 Kb buffer size with no performance degradation.

Also, be sure that mTape Server is installed with a buffer size equal to the record size in your tape configuration file. If it is not, that extra space is wasted.

If the AdminAgent is installed on a different board from mTape Server, you need buffers. Assume also that you can spare the memory for 80 Kb worth of buffers. You could allocate 20 4–Kb buffers or four 20–Kb buffers. The tape configuration file states the record size.

The difference between many small buffers and a few large buffers is that it takes longer to fill up the large buffers. Your goal is to keep the tape moving (streaming). mTape Server does not write partial buffers, so it would probably be waiting for the 20 Kb buffers to fill up instead of continuously writing tape records. The 4 Kb buffers fill up quickly. Since mTape Server can fill a buffer while simultaneously writing to tape, but the time one small buffer is written, the next buffer is already full and ready to be written.

Ideally, only two buffers should be needed, and they should be large enough so that it takes as long to write one buffer to tape as it does to fill the other buffer from disk. This exact match is almost impossible, so you should experiment.

First, remove all other servers from the DP or SP. Then, allocate a generous amount of large buffers. The tape should stream when you are doing a backup. Gradually decrease the size of the buffers until the tape no longer streams. Then use the smallest size at which the tape still streams. Repeat the process, decreasing the amount of buffers. Remember to allocate more, rather than fewer, buffers so that you won't have performance problems if the system load changes.

Do not be concerned if the Restore operation does not stream. Backup (reading from disk and writing to tape) is generally faster than Restore (reading from tape and writing to disk). During the Restore operation, the tape utilities not only have to write the data to disk, they have to unformat the disk records from the tape record, create directories when necessary, update entries in Mfd.sys and FileHeaders.sys, etc. By contrast, reading from disk during backup is done quickly enough to keep mTape Server busy.

Consider, also, that the mTape Server writes an interrecord gape of 3/4 of an inch. This means that the smaller the records, the more interrecord gaps there will be on the tape. However, this should not be a problem is you use 2400–foot tapes.

Configuring System Services

You create the standard set of XE520 BTOS system configuration files by using the mBTOS CONFIG utility. These files define your system's hardware and software configuration.

The mBTOS CONFIG utility provides a menu–driven facility that allows you to modify standard system configuration files.

Note: Sections 6 and 7 provide overviews of BTOS system configuration files and system services. You should read these sections before configuring your system with the mBTOS CONFIG utility.

Master Configuration File

The master configuration file allows you to:

- 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

The processor initialization files allow you to:

- 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

The processor configuration files allow you to:

- 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

When to Use the mBTOS CONFIG Utility

You can use the mBTOS CONFIG utility to:

- create the system configuration files if you are installing XE520 BTOS software for the first time
- modify current files or create new files to reflect changes in your system configuration if you have already installed and configured system software

Note: Until you become familiar with the configuration of your system, you must create the new processor's configuration file using the mBTOS CONFIG utility.

Creating System Configuration Files

Note: If you have initialized the system disk and have just completed installing system software, do not reboot the system. Leave the system in the restricted mode until you have created the system configuration files with the mBTOS CONFIG utility.

Each processor in the system must contain 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 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 non-default 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.

To create the BTOS system configuration files, complete the following:

1 Type **mBTOS Config** at the BTOS Executive command line.

2 Press **GO**.

The mBTOS CONFIG utility begins running. The system creates the master configuration file and displays its contents. The system displays the Default Menu and prompt following the master configuration file listing.

3 At the Default Menu prompt, type **2**.

4 Press **RETURN**.

The utility creates default configuration and initialization files for each processor listed in the master configuration file, and then lists these files.

After the processor configuration and initialization files are created, the system displays the message **mBTOS Config Utility Complete** and returns you to the BTOS Executive command line.

5 Review all processor initialization files to make sure that they match the actual system configuration.

6 If you need to modify any files to match the system configuration, run the mBTOS CONFIG utility again, selecting the **Do Not Use Default** option of the mBTOS CONFIG Default Menu.

If the BTOS system configuration files do not need to be modified, or when you have finished modifying them, proceed to Rebooting the System after Running mBTOS CONFIG, later in this section.

Starting the mBTOS CONFIG Utility

To invoke the mBTOS CONFIG utility, type **mBTOS Config** at the BTOS Executive command line and press **GO**.

Each mBTOS CONFIG menu comprises the following:

- 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 **RETURN** or **GO** to execute the action.

When you invoke the mBTOS CONFIG utility, the system displays a welcome message, along with a listing of the standard master configuration file ([sys]<sys>Master.cnf).

If the master configuration file does not exist, the system creates it dynamically when you invoke the mBTOS CONFIG utility. The utility polls the actual processor boards installed in the system to find out what processors exist. 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, the system displays its contents when you invoke the mBTOS CONFIG utility.

After displaying the listing of the master configuration file, the system displays 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

Creating the Default Versions of All Files

The system displays the main Default Menu and prompt as shown in figure 8–1.

To create the default files, select option 2 and press **RETURN**. The utility creates and lists default configuration and initialization files for each processor listed in the master configuration file.

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.

A default initialization file and default configuration file exist for each type of processor. Their file names take the following form:

[sys]<sys>DefprocID.cnf

or

[sys]<sys>DefInitprocID.jcl

ID represents the appropriate processor name and number (refer to appendix A).

After the processor configuration and initialization files are created, the system displays the message **mBTOS Config Utility Complete** and returns you to the BTOS Executive command line.

Modifying the Master Configuration File

The system displays the main Default Menu and prompt as shown in figure 8–1.

Figure 8–1 **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, select option 1 and press **RETURN**. The system displays the Main Menu, from which you can select the type of file you wish to create or modify.

The system displays the Main Menu and prompt as shown in figure 8–2.

To modify the master configuration file, select option **1** and press **RETURN**. The system displays a listing of the master configuration file, 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 shown in figure 8–3.

Figure 8–2 **Main Menu**

```
0 = EXIT UTILITY
1 = MASTER CNF FILE
2 = JCL FILE
3 = CNF FILE
Please Select Type of File ==>
```

Figure 8–3 **Function Menu**

```
0 = RETURN TO MAIN MENU
1 = ADD
2 = DELETE
Please Select a Function for the Master CNF File ==>
```

Adding Processor Entries

To add the appropriate entries for a processor, select option 1 in response to the Function Menu prompt and press **RETURN**. The system displays the Add Menu, from which you can select the type of processor that you have added to the system.

The system displays the Add Menu and prompt as shown in figure 8–4.

Select the number that corresponds to the type of processor you have added to the system and press **RETURN**. The system displays the Add Board Menu and prompt. The Add Board Menu allows you to select the location in the master configuration file at which the processor entries should appear.

The Add Board Menu lists each of the processors that contain entries in the master configuration file. In between the processor name are numbered locations at which the new processor entries can be added.

The system displays the following prompt after the Add Board Menu:

Please Select the Location to Add the Processor Board
==>

Select the number that corresponds to the appropriate location and press **RETURN**.

Figure 8–4 **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 ==>
```

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 XE520 enclosure(s) (that is, by slot number order).

After a location has been selected, the system displays the updated master configuration file, and returns the Function Menu to allow further modifications.

When updating the master configuration file, the mBTOS CONFIG utility adds the appropriate entries to the master configuration file. If the processor being added requires an include statement for its configuration file, the mBTOS CONFIG utility automatically adds the include statement. The mBTOS CONFIG utility also updates the initialization and configuration file names for any processors that follow the new processor and are of the same type. (For example, if a you add a new CP to the system between CP01 and CP02, the mBTOS CONFIG utility automatically renames files related to CP02 to reflect its new number, CP03.)

Note: Until you become familiar with the configuration of your system, you must create the new processor's configuration file using the mBTOS CONFIG utility.

Deleting Processor Entries

To delete the appropriate entries for a processor, select option 2 in response to the Function Menu prompt and press **RETURN**. The system displays the Delete Board Menu, 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. The system displays the following prompt after the Delete Board Menu:

Please Select Processor Board to Delete from the Master Cnf ==>

Select the number that corresponds to the board that has been removed from the system and press **RETURN**. The system displays the updated master configuration file, and returns the Function Menu to allow for further modifications.

When updating the master configuration file, the mBTOS CONFIG utility deletes the appropriate entries to the master configuration file. If the processor being deleted had an include statement for its configuration file, the mBTOS CONFIG utility automatically deletes the include statement. The mBTOS CONFIG utility also updates the initialization and configuration file names 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 you removed CP01, the mBTOS CONFIG utility automatically renames files related to CP02 to reflect its new number, CP01.)

Selectively Creating or Modifying Processor Initialization Files

Note: Before you attempt to modify processor initialization files, you must know the restrictions regarding where you can run system services, how many copies of a given system service you can run, and so on. For more information, refer to Guidelines for Running System Services, later in this section.

The system displays the main Default Menu and prompt as shown in figure 8–1.

To create or modify a processor initialization file, select option 1 and press **RETURN**. The system displays the Main Menu, from which you can select the type of file you want to create or modify.

The system displays the Main Menu and prompt as shown in figure 8–2.

Note: The mBTOS CONFIG utility refers to processor initialization files as JCL files.

To modify a processor initialization file, select option 2 and press **RETURN**.

The system displays the Board Menu, which allows you to choose which processor's initialization file you want to create or modify. The mBTOS CONFIG utility examines the master configuration file to generate the list of processor boards.

The system displays the following prompt after the Board Menu:

Please Select a Specific JCL file ==>

Select the number corresponding to the processor board whose initialization file you want to create or modify and press **RETURN**. 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 system displays the JCL Default Menu and prompt as shown in figure 8–5.

Creating the Default Version

To create the default version of the processor initialization file, select option 2 and press **RETURN**. The system creates and displays the default initialization file.

Note: The mBTOS CONFIG 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, the mBTOS CONFIG utility opens a new file before the copy is made. If the processor initialization file does exist, it is overwritten during the copy operation.

After the system displays the new initialization file, the mBTOS CONFIG utility returns to the Main Menu.

Figure 8–5 **JCL Default Menu**

```
0 = RETURN TO MAIN MENU
1 = DO NOT USE DEFAULT
2 = USE DEFAULT JCL FILE

Please Select Mode of Operation ==>
```

Modifying an Existing File or Creating a Nondefault File

To modify the existing processor initialization file or to create a nondefault one, select option 1 in response to the JCL Default Menu prompt and press **RETURN**. If the file already exists, the system displays its contents. If the file is new, an empty file results.

After the system lists the initialization file, it displays a Function Menu. The Function Menu allows you to choose whether you want to add or delete a system service entry.

The system displays the Function Menu and prompt as shown in figure 8–6.

To add a system service to the file, select option 1 and press **RETURN**. The system displays an Add Server Menu and prompt, which menu lists the system services that can be run on the processor.

The system displays the following prompt after the Add Server Menu:

Please Select a Server to Add to the JCL file ==>

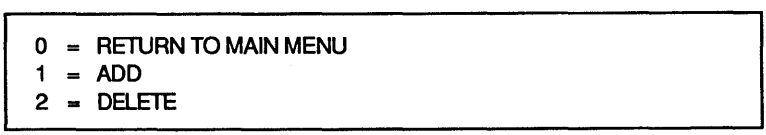
Select the number corresponding to the service you wish to add and press **RETURN**. The system displays the updated file, and then displays the Function Menu and prompt to allow for further modifications.

To delete a system service from the file, select option 2 and press **RETURN**. The system displays a Delete Server Menu and prompt, which lists the system services that are currently to be run on the processor.

The system displays the following prompt after the Delete Server Menu:

Please Select a Server to Delete from the JCL file ==>

Figure 8–6 **JCL Function Menu**



Select the number corresponding to the service you wish to delete and press **RETURN**. The system displays the updated file, and then displays the Function Menu and prompt to allow for further modifications.

When you are finished modifying the file, select option 0 in response to the Function Menu prompt to return to the Main Menu.

Selectively Creating or Modifying Processor Configuration Files

The system displays the main Default Menu and prompt as shown in figure 8–1.

To create or modify a processor configuration file, select option 1 and press **RETURN**. When the system displays the Main Menu, you can select the type of file you wish to create or modify.

The system displays the Main Menu and prompt as shown in figure 8–2.

Note: The mBTOS CONFIG utility refers to processor configuration files as CNF files.

To modify a processor configuration file, select option 3 and press **RETURN.**

The system displays the Board Menu, which allows you to choose which processor's configuration file you want to create or modify. The mBTOS CONFIG utility examines the master configuration file to generate the list of processor boards.

The system displays the following prompt after the Board Menu:

Please Select a Specific CNF file ==>

Select the number corresponding to the processor board whose configuration file you want to create or modify, and press **RETURN**. The CNF Default Menu appears and 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 BTOS Editor (refer to section 6).

The system displays the CNF Default Menu and prompt as shown in figure 8–7.

Creating the Default Version

To create the default version of the processor configuration file, select option 2 and press **RETURN**. The system creates and displays the default configuration file.

Note: The mBTOS CONFIG 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, the mBTOS CONFIG utility opens a new file before the copy is made. If processor configuration file does exist, it is overwritten during the copy operation.

After the system displays the new configuration file, the mBTOS CONFIG utility returns to the Main Menu.

Figure 8–7 **CNF Default Menu**

```
0 = RETURN TO MAIN MENU
1 = DO NOT USE DEFAULT
2 = USE DEFAULT CNF FILE

Please Select Mode of Operation ==>
```

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, select option 1 in response to the CNF Default Menu prompt and press **RETURN**. If the file already exists, the system displays its contents. If the file is new, an empty file results.

After the system lists the configuration file, it displays a Function Menu, which allows you to choose whether you want to add or delete a port definition entry.

The system displays the Function Menu and prompt as shown in figure 8–3.

The following subsections describe the procedures for adding or deleting entries to TP and CP configuration files

---

Caution: A CP configuration file must contain 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, the system tells you 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, select option 1 in response to the Function Menu and press **RETURN**. The system displays a prompt that allows you select the port entry you want to add or to return to the Function Menu. The system displays the following prompt:

*** Valid answers are numbers between 1 and 10 ***
Press 0 to Return or Insert a Port Number to Configure
==>

Select the number corresponding to the RS–232–C port entry you wish to add and press **RETURN**. The system displays the following:

The Current Line Being Added
async n, speed =1200, parity =none, stopbits =1, charbits=8

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 system displays the Change Line Menu and prompt as shown in figure 8–8.

To add the port entry, as displayed, to the TP configuration file, select option 0 and press **RETURN**. After the entry is added, the system displays the updated TP configuration file, and the Add Menu and prompt to allow further modifications.

To cancel the addition of the port entry, select option 1 and press **RETURN**. The entry is not added, and the system displays the port selection prompt to allow you to select another port.

To change the line speed defined in the port entry, select option 2 and press **RETURN**. The system displays the Speed Menu and prompt, which 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.

Figure 8–8 **Change Line Menu**

```
0 = RETURN TO ADD MENU
1 = PORT
2 = SPEED

Please Select the Parameter to Change ==>
```

The system displays the following prompt after the Speed Menu:

Please Select a Speed ==>

Select the number corresponding to the desired speed and press **RETURN**. The system displays the modified port entry, 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, the system tells you 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, select 1 in response to the Function Menu and press **RETURN**. The system displays an Add Menu, which lists the CP's port types. The system displays the following prompt after the Add Menu:

Please Select an Item to Configure ==>

Select the number corresponding to RS–232–C and press **RETURN**. The system displays a Port Menu listing the CP's RS–232–C ports.

The system displays the following prompt after the Port Menu:

Please Select a Port to Configure ==>

Select the number corresponding to the port entry you wish to add or modify and press **RETURN**. The system displays the following:

The Current Line Being Added
async n, speed =1200, parity =none, stopbits =1, charbits =8

n is the port number that you selected. After the system displays the port entry, it displays 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 system displays the Change Line Menu and prompt as shown in figure 8–8.

To add the port entry, as displayed, to the CP configuration file, select option 0 and press **RETURN**. 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, select option 1 and press **RETURN**. 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, select option 2 and press **RETURN**. The system displays the Speed Menu and prompt, which 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 system displays the following prompt after the Speed Menu:

Please Select a Speed ==>

Enter the number corresponding to the desired speed and press **RETURN**. The system displays the modified port entry, 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, the system tells you 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 **RETURN**. The system displays an Add Menu, which 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 **RETURN**. The system displays a Line Menu 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 **RETURN**. The system displays the following:

The Current Line Being Added
line n, clock =307k

n is the line number that you selected. After the system displays the line entry, it 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 system displays the Change Line Menu and prompt as shown in figure 8–8.

To add the line entry, as displayed, to the CP configuration file, enter **0** and press **RETURN**. The entry is added, the updated CP configuration file is displayed, and the Add Menu and prompt are redisplayed to allow further modifications.

To cancel the addition of the line entry, enter **1** and press **RETURN**. 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 **RETURN**. The system displays the Speed Menu, which 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 **RETURN**. The system displays the modified line entry, 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.

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 **RETURN**. The system displays a Delete Line Menu, which 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 want to delete and press **RETURN**. The system displays theupdated file, and then displays the Function Menu to allow 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 **RETURN**. If the file already exists, the system displays its contents. If the file is new, an empty file results.

The system lists the configuration file, followed by a Function Menu. The Function Menu allows you to choose whether you want to add or delete an entry.

The system displays the Function Menu and prompt as shown in figure 8–3.

Enter **1** in response to the Function Menu and press **RETURN**. The system displays an Add Menu, which 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 **RETURN**. The system displays the Block Menu and prompt, which 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 ==>

n is the block type you selected.
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 **RETURN**. You must enter a three-digit number (for example, to specify 12 blocks, enter **012**). The mBTOS CONFIG 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 mBTOS CONFIG utility returns to the Block Menu.

Guidelines for Running System Services

The mBTOS CONFIG utility allows you to install system services on specific processor boards. Table 8–1 lists the available system services, their corresponding run file names, and the amount of memory they require.

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 8–2 lists the system services and applicable restrictions.

If you receive the BTOS error code 400 (**Out of Processor Memory**), use the PLOG command 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 add a Memory Expansion (ME) board to the processor or move system services to another processor.

Table 8–1 **XE520 BTOS System Services and Run Files**

System Service	Run File	Memory
AdminAgent	[sys]<sys>MFAdminAgent.run	64 Kb
Queue Manager	[sys]<sys>mInstallQMgr.run	34 Kb
Printer Spooler Manager	[sys]<sys>mSpoolerMgr.run	70 Kb
Tape Server	[sys]<admin>mTapeServer.run	180 Kb
QIC Server	[sys]<admin>mQicServer.run	100 Kb

Table 8–2 **Restrictions for XE520 BTOS System Services**

System Service	Runs On	Number	Partition
AdminAgent	Any	One per processor	Primary
Queue Manager	Any	One per system	Secondary
Printer Spooler Manager	CP, TP	One per CP or TP	Primary
Tape Server	DP or SP	One per system	Secondary
QIC Server	Any	One per system	Secondary

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.

If you have just created your system configuration files as part of the software installation procedure, reboot the system. Then proceed to section 9 for instructions on performing recommended post–configuration tasks.

Post–Configuration Tasks

After creating the system configuration files, it is recommended that you perform the following tasks:

- create BTOS command forms and files
- get a file and printed listing of the bad spots for all disks in your system
- restore any customized files that you saved (refer to section 4)
- backup your system software

BTOS Command Forms and User Command Files

The run files for BTOS utilities are installed as part of the BTOS software installation procedure. However, the command forms for most of these utilities do not exist until appropriate submit files are run, using the SUBMIT command. These submit files run a NEW COMMAND command for each BTOS command included in the submit file, thereby creating the command forms.

The submit files also copy the commands into a user command file. Through an entry in the user's signon file, each user is assigned a command file. This command file contains the commands to which a user has access. For more information about how command files are used, refer to your *XE500 BTOS Administration Guide*.

By default, commands are copied into the command file [sys]<sys>Sys.cmds. This command file is used when you log onto the system without entering a user name in the Signon form (that is, the user defined by the user signon file [sys]<sys>.user).

During the software installation procedure, the workstation commands are copied into [sys]<sys>Sys.cmds. You must now create and copy the XE520 commands into the default command file and/or user command files.

Creating the Command Forms and Command Files

To create the XE520 BTOS command forms and add them to a command file, complete the following:

1 Copy the file [!sys]<sys>Sys.cmds to the user command files you wish to create.

   At this point, the command file [!sys]<sys>Sys.cmds contains all of the workstation commands and the XE520 master commands that were added during the installation procedure.

2 Type **Submit** at the BTOS Executive command line.

3 Press **RETURN**.

   The SUBMIT command form is displayed on the workstation screen.

4 In the **File List** field, type the submit file name for the command group whose forms you want to create.

   Refer to table 9–1 for a listing of the command group submit files and the commands created by them.

   To create all of the available commands, use the wild card character (*) in the submit file specification, (for example, [!sys]<backup>*.sub).

5 In the **[Parameters]** field, type the name of the user command file to which you want the commands added. If you want to add the commands to [!sys]<sys>Sys.cmds, leave this field blank.

6 When you have completed filling in the SUBMIT command form, press **GO**.

   The operations that are required to create the commands are displayed at the screen. When all of the commands have been created, the BTOS Executive command line is displayed.

7 If there are any commands in a user command file that you want to remove, use the REMOVE COMMAND command. (For example, you may not want a user to have access to mIVOLUME, mBTOS CONFIG, or other commands with which a user could corrupt system files.)

Repeat this procedure for each group of commands and each command file you wish to create.

Recording System Disk Bad Spots

If you ever have to reinitialize your system disk, you will need a listing of the disk's bad spots.

You use the BAD SPOT REPORT command to list the information about the bad spots on your system disk. In the BAD SPOT REPORT command form, use the **[Print file]** option to copy this information to a printable file. Print the file and save the hardcopy listing and the files of your system disk bad spots in a safe place.

Refer to your *XE500 BTOS Administration Guide* for details on BAD SPOT REPORT and bad spot information.

Listing Disk Bad Spots

In the event that a disk needs to be reinitialized, you will have to enter a listing of known bad spots in the MIVolume command form. Having a file listing of these bad spots makes it easier to enter the bad spots. Also, if a disk's volume control structure bad spot listing becomes corrupted, a printed copy of the listing will be the only record of known bad spots on the disk.

For the procedure to generate a file and printed listing of disk bad spots, refer to the *XE500 BTOS Administration Guide.*

Restoring Customized Files

Once the system software has been installed, you can reinstall any customized system files that you saved.

The following are the only types of customized system files that you can restore:

- device configuration files (files created through the mCREATE CONFIGURATION FILE utility)
- the queue index file, [sys]<sys>queue.index
- the printer spooler configuration files (for example, [sys]<spl>splcnfg.sys (refer to section 11)).

Backing Up System Software

If your system has a hardware failure, it may be impossible to access data stored on a disk. To minimize the impact of such a failure, you should backup (archive) the system files onto half–inch tapes or QIC tapes immediately after completing the system configuration procedures.

In the event that a system failure prevents access to file data, you can restore the archived system files from your tapes.

For information about archiving procedures, refer to the *XE500 BTOS Administration Guide*.

Configuring the Queue Manager

The Queue Manager 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.

The Queue Index File

The 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:

- the queue's name (to be used by the system)
- the queue entry file to hold the queue's requests
- the size of each queue entry
- the queue type

Each queue entry has the following format:

QueueName/FileSpec/EntrySize/QueueType

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 table 10–1 (Unisys reserves type 0–80; values may range from 0–255):

Table 10–1 **Queue Types**

Queue Type	Assignment
1	Printer spooler queue
2	BSC RJE queue
3	Batch queue
4	SNA RJE queue

Installing the Queue Manager

To install the Queue Manager, complete one of the following:

- Execute the mINSTALL QUEUE MANAGER command from the BTOS Executive.
- Include the Queue Manager run statement in your processor's initialization file. This causes the Queue Manager to be installed automatically at system start–up.

When running the Queue Manager, remember the following:

- You can have only one XE520 Queue Manager installed and running at a time.
- The Queue Manager can run on any processor.
- The Queue Manager uses 34 Kb of processor memory. It could use more, depending on whether caching is used and on the number of queues it controls.

To execute the mINSTALL QUEUE MANAGER command, complete the following:

1. Type **mInstall Queue Manager** at the BTOS Executive command line.
2. Press **RETURN**.

   The system displays the command form shown in figure 10–1. Table 10–2 lists the optional fields in the mINSTALL QUEUE MANAGER command.
3. When you have filled in the appropriate fields of the mINSTALL QUEUE MANAGER command form, press **GO**.

   The system prompts you for the processor on which the partition resides.
4. Enter the appropriate processor name (for example, FP00, CP01, and so on) after the prompt.
5. Press **RETURN**.

   The Queue Manager is installed into the selected processor's primary partition as specified.

Figure 10–1 **mINSTALL QUEUE MANAGER Command Form**

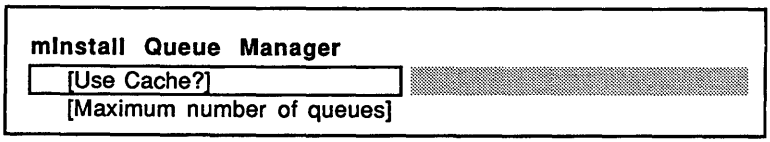


Table 10–2 **mINSTALL QUEUE MANAGER Command Fields**

Field	Action/Explanation
[Use Cache?]	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. Leave this field blank if you do not want the Queue Manager installed with an extra buffer area.
[Maximum number of queues]	Enter 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.

To have the Queue Manager installed automatically at system start–up, use the mBTOS CONFIG utility to add the Queue Manager run statement in the desired processor's initialization file. The Queue Manager run statement is: $run [sys]<sys>mInstallQMgr.run.

The default parameters, discussed in table 10–2, are used when the Queue Manager is installed at system start–up, unless you manually edit the run statement to include nondefault parameters. (Refer to the *XE500 BTOS Administration Guide* for information about including run file parameters in CLI run statements.) At system start–up, the Queue Manager initially executes in the processor's primary partition, but installs itself into a secondary partition.

Stopping the Queue Manager

To stop the Queue Manager from running, regardless of how it was installed, use the mDEINSTALL QUEUE MANAGER command.

To execute the command, complete the following:

1. Type **mDeInstall Queue Manager** at the BTOS Executive command line.
2. Press **GO**.

   The system prompts you for the processor on which the Queue Manager is installed.
3. After the prompt, enter the appropriate processor name (for example, FP00, CP01, and so on) after the prompt.
4. Press **RETURN**.

   The Queue Manager is deinstalled.

If you are running multiple AdminAgents and have a subordinate 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. Use the mDEINSTALL QUEUE MANAGER command to stop the Queue Manager from running.
3. Remove the queue entry file that corresponds to the problem print queue.
4. Use the mINSTALL QUEUE MANAGER command to reinstall the Queue Manager on the processor.

Queue Requests

The following is a list of requests supported by the XE520 Queue Manager:

AddQueue
ApplicationSwapped
ApplicationTerminated
DeInstallQueueManager
GetQMStatus
RemoveQueue

For information about other queue requests, refer to your Operating System reference documentation.

Configuring the 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.

The printers connected to an XE520 processor board (either a CP or TP) are controlled by a printer spooler manager running on that processor. A printer spooler manager can also run locally at a workstation to control any printers connected to the workstation. Regardless of where spooler managers are run, all spooled printing queues in the system are controlled by the XE520 Queue Manager.

Note: If you are running the Generic Print Service (GPS), refer to your GPS documentation.

XE520 files to be printed must be sent to a print queue associated with the desired printer.

The default 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 print queue called spl.

- a serial printer connected to channel 3 of CP00

  The serial printer name is SERIAL and it services the print queue called splb.

Operating the Printer Spooler

A 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 printer spooler can resume automatically after a BTOS reboot.

There is no one file that can be referred to as the printer spooler file. A printer spooler actually contains:

- a set of printer spooler queues, which are under the control of the Queue Manager
- a 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 spooler manager to be installed in the processor at boot time.
- line entries in the CP or TP configuration file for each RS-232-C port to which a printer is connected

  The line characteristics defined in this entry are overridden by those defined in the printer's configuration file. However, the entry in the processor configuration file is still required.

Printer Spooler Queues

The Queue Manager controls the following queues related to printer spooler functions:

- a scheduling queue corresponding to each print queue
- a control queue for every printer that serves a printer spooler
- one spooler status queue, regardless of the number of printer spoolers

Sample Printer Spooler Queue Entries

As an example, a queue index file could contain 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 define 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 one sector, and the queue to be a printer spooler–related queue.

The Spooler Configuration File

There is one spooler configuration file, [sys]<sys>splcnfg.sys, for all printer spoolers installed on XE520 processors. The spooler configuration file contains an entry for each printer being used with the printer spoolers. A printer's entry defines:

- the processor channel to which the printer is connected
- the printer's name
- the print scheduling queue that serves that printer
- the printer's configuration file
- the printer's service priority level
- whether a banner is to be printed before each print job sent to the printer
- 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

ChannelName is a single character abbreviation that specifies the channel to which the printer is connected. Table 11–1 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.

PrinterName is the arbitrary name of the printer. All 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. If you specify Yes, a banner is printed at the beginning of each file; if you specify No, no banner page is printed.

Processor is the name of the processor board to which the printer is connected (for example, CP00, TP01).

Table 11–1 **CP and TP Printer Channels**

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)

Sample Spooler Configuration File

The following sample 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
```

The first entry tells the following information about its corresponding printer:

- the printer is connected to the parallel printer channel
- the printer's name is PARALLEL

  The printer's control queue name in the queue index file is PARALLELcontrol.
- the printer services the SPL scheduling queue
- the printer's configuration file name is [sys]<spl>SPLconfigCp00.sys
- the printer's priority level is 64
- a banner will be printed for each printing task sent to the printer
- 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 contains specific information about a printer's operating parameters. A configuration file for a printer can be created or modified by using the mCREATE CONFIGURATION FILE command.

Serial Printer Channel Entries

When a serial printer is attached to a CP or TP, the processor's configuration file must contain a line entry for the RS–232–C channel to which the printer is connected. This entry must define:

- if the channel is using sync or async mode
- the line speed to be used
- the parity mode, number of stop bits, and number of character bits being used for data transfer

An RS–232–C channel line entry may also contain other optional parameters. Refer to section 6 for the format of these line entries.

Line speed, parity, stopbits, and charbits parameters in any async RS–232–C entries are ignored by the printer spooler. They are required by the CP and TP to configure the channel as an asynchronous channel during system boot. Therefore, they must be present. However, the printer spooler reconfigures the channel when it starts to run according to the configuration values supplied in the configuration file of the printer attached to the channel.

What the Printer Spooler Does

The following types of operations are controlled through the printer spooler:

- requesting that a file be printed
- controlling an active printing task (for example, cancelling the task, halting the task, resuming the task after a halt)
- requesting spooler status information

When a print request is made, the request is placed in the specified scheduling queue (that is, the queue specified in the PRINT command or the SPOOLER STATUS command). 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, regardless of whether a specific printer has been specified in the SPOOLER STATUS command. 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 by means of the SPOOLER STATUS command on a particular printer, 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 SPOOLER STATUS command.

Modifying the Printer Spooler Operation

When you want to change your system's 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, complete the following:

- Use the mCREATE CONFIGURATION FILE command to create the printer configuration file for the printer.
- Use the BTOS Editor to add an entry for the printer in the spooler configuration file.
- Use the BTOS Editor to add an entry for the printer's control queue to the queue index file.

- If the printer is to have its own schedule queue, use the Editor to add an entry for that schedule queue to the queue index.
- If adding a serial printer, use the mBTOS CONFIG utility to add the appropriate async entry to the processor'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, regardless of whether a printer has been specified in the SPOOLER STATUS command.

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), complete the following:

- Be sure that the CP or TP to which the printer is to be connected is running a printer spooler manager.
- Modify the printer's entry in the spooler configuration file to reflect the printer's new location and/or scheduling queue.
- 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.

  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.

Running a Workstation Printer Spooler

There must be one spooler configuration file for each workstation printer spooler. These printer spoolers are under the control of the XE520 Queue Manager. Note that to obtain information about the status of the workstation's printer spooler, you must use the SPOOLER STATUS command.

Refer to your Standard Software documentation for information about configuring the printer spooler operation at a workstation. Keep in mind that the XE520 is considered the master workstation and that the queue manager for the workstation resides at the XE520.

Printers connected to workstations that are running printer spooler are available to all users through the XE520 Queue Manager.

Printer Translation Files

The MAKE 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 MAKE 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. Therefore, you should not use translation files for text substitution.

Use the BTOS Editor to create a source file that lists the characters you want to translate (refer to your BTOS Editor documentation). When the source file is complete, use the MAKE TRANSLATION FILE command to convert the source file into a translation file. You can then specify the translation file when you create or modify the printer device configuration file.

Creating the Source File

To create a source file, use the Editor. Make an entry for each character you want to translate, using ASCII hexadecimal (hex) codes to represent the characters (refer to the ASCII table in your Standard Software documentation). Entries should be in the following format:

XX = YY

XX is the ASCII hex code for the character you want to translate.

YY is the ASCII hex code for the translated character.

For example, if you want 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 lowercase letters to uppercase is included with the standard XE520 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 MAKE TRANSLATION FILE command to convert a source file into a translation file.

To execute the MAKE TRANSLATION FILE command, complete the following:

1 Type **Make Translation File** at the BTOS Executive command line.

2 Press **RETURN**.

The system displays the command form shown in figure 11–1. Table 11–2 describes the two command fields.

3 Enter a parameter in both fields.

4 Press **GO**.

The system creates the translation file from the source file specified.

Figure 11–1 **MAKE TRANSLATION FILE Command Form**

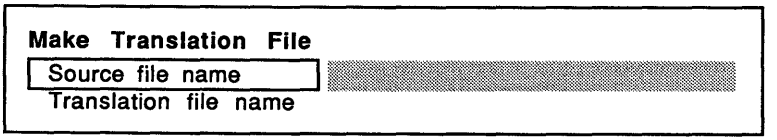


Table 11–2 **MAKE TRANSLATION FILE Command Form Fields**

Field	Action/Explanation
Source file name	Enter the name of the source file you want to convert into a translation file.
Translation file name	Enter the name of the translation file.

Creating I/O Device Configuration Files

XE520 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 6 for a description of the default I/O device configuration files.

To create or modify a device configuration file, you use the mCREATE CONFIGURATION FILE utility.

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 following printer configuration file name was selected for the printer connected to channel C (that is, channel 3) of processor TP00:

[sys]<spl>SPLCConfigTp00.sys

The characters SPLC are included to indicate that the file is a spooler–related configuration file. The term .sys is appended to indicate a system service file. The file is stored in the directory <spl> with other spooler–related files.

Executing the mCREATE CONFIGURATION FILE Utility

The mCREATE CONFIGURATION FILE command invokes a menu-driven utility through which a configuration file for an XE520 I/O device can be created. I/O devices include serial printers, parallel printers, tape drives, and modems.

The mCREATE CONFIGURATION FILE command does not have a command form. The menu-driven facility is invoked by entering the command name at the BTOS Executive command line and pressing **GO**.

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

To invoke the mCREATE CONFIGURATION FILE utility, complete the following:

1 Type **mCreate Configuration File** at the BTOS Executive command line.

2 Press **GO**.

The following prompt appears:

Enter Configuration File Name:

3 Type the name of the new configuration file to be created, or the name of an existing file to be modified.

4 Press **RETURN**.

The following prompt appears:

Tape, QIC, Parallel Printer, Serial Printer, Communications:

5 To 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 (such as, t, q, p, s, or c).

6 Press **RETURN**.

The following prompt 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.

7 To advance to the next prompt, press **RETURN**.

If you press **RETURN** after the last prompt, you loop back to the first prompt.

8 When you have entered the values you want for all of the prompts, press **GO**.

The following prompt appears:

Continue to Configure?

9 Press **GO** to create or modify another configuration file.

The system again prompts you for input.

10 If you do not want to create or modify another configuration file, press **FINISH** to exit the mCREATE CONFIGURATION FILE utility.

Device Type Prompt

A different set of prompts is displayed for each device type. The following subsections describe 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) or **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 [2048]	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 [Yes]	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 XE520 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 11 for a description of the MAKE TRANSLATION FILE command 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 XE520.

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 11 for a description of the MAKE TRANSLATION FILE command 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**). The default value, 65535, causes a timeout of 0 seconds (that is, an immediate return). Specifying the value 0 causes an infinite time-out (that is, the read operation waits indefinitely).
End of File Byte--None or Hex Value [04]	Enter the hex value which indicates 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 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 error code 9080 is reported.

If you get this error code when running an 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.

Appendix A

Default System Configuration Files

mBTOS CONFIG General Default Initialization Files

[sys]<sys>DefInitFb00.jcl (Master FP Systems with QIC Tape)
$continueonerror
$run [sys]<sys>mInstallQMgr.run
$run [sys]<admin>mQicServer.run,85
$run [sys]<sys>MfAdminAgent.run

[sys]<sys>DefInitCb00.jcl
$run [sys]<sys>mSpoolerMgr.run

[sys]<sys>DefInitDb00.jcl (Master DP Systems without QIC Tape)
$continueonerror
$run [sys]<sys>mInstallQMgr.run
$run [sys]<admin>mTapeServer.run,2,16
$run [sys]<sys>MfAdminAgent.run

[sys]<sys>DefInitDb00.jcl (Master DP Systems with QIC Tape)
$continueonerror
$run [sys]<sys>mInstallQMgr.run
$run [sys]<admin>mQicServer.run,85
$run [sys]<sys>MfAdminAgent.run

[sys]<sys>DefInitDb00.jcl (Master FP Systems)
$run [sys]<admin>mTapeServer.run,2,16

[sys]<sys>DefInitSb00.jcl
$run [sys]<admin>mTapeServer.run,2,16

[sys]<sys>DefInitTb00.jcl
[This file is empty.]

mBTOS CONFIG General Default Configuration Files

[sys]<sys>DefFb00.cnf (Master FP Systems)
BlockType=y, Number=23
BlockType=z, Number=93
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>DefFb01.cnf (Master FP Systems)
BlockType=y, Number=23
BlockType=z, Number=95
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>DefFb02.cnf (Master FP Systems)
BlockType=y, Number=23
BlockType=z, Number=95
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>DefFb03.cnf (Master FP Systems)
BlockType=y, Number=23
BlockType=z, Number=95
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>DefDb00.cnf
BlockType=y, Number=23
BlockType=z, Number=95
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>DefDb01.cnf
BlockType=y, Number=23
BlockType=z, Number=95
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>DefCb00.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>DefSb00.cnf
BlockType=y, Number=4
BlockType=z. Number=30

[sys]<sys>DefTb00.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
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
End
```

[sys]<sys>InitFp00.r.jcl (Master FP Systems with QIC Tape)

```
$continueonerror
$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>mQicServer.run,85
$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]<admin>mCreateSwapR.run
$run [sys]<admin>mCreatePartition.run,150k,UFS
$run [sys]<admin>mInstallServer.run,UFS,
   [sys]<sys>UFS.run
$run [sys]<admin>mTapeServer.run,2,16
$run [sys]<sys>mInstallQMgr.run
$run [sys]<sys>MfAdminAgent.run
```

[sys]<sys>InitCp00.r.jcl (Master FP Systems, Master DP Systems without QIC Tape)

```
$run[sys]<sys>mSpoolerMgr.run
```

[sys]<sys>InitCp00.r.jcl (Master DP Systems with QIC Tape)

```
$run [sys]<admin>mQicServer.run,85
$run [sys]<sys>mSpoolerMgr.run
```

[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

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>IVAtasi46.cnf:
; THIS IS FOR AN ATASI 46 MB DRIVE
CYLINDERS=645
TRACKS=7
SECTORS=16
BYTES=512
STEP=0
REMOVABLE=no
END

[sys]<sys>IVMicropolis85.cnf:

```
; THIS IS FOR A MICROPOLIS 85MB DRIVE
CYLINDERS=1024
TRACKS=
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
```

Appendix B

Hardware Configuration Information

Figure B–1 **Built–in Disk Device Naming Conventions**

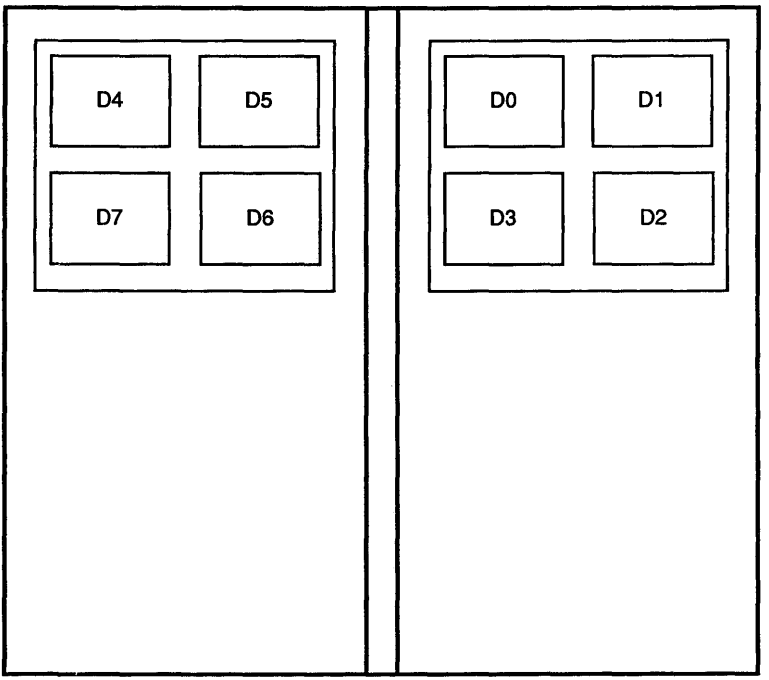


Figure B–2 **Built–in Disk Device Naming Conventions for DP00**

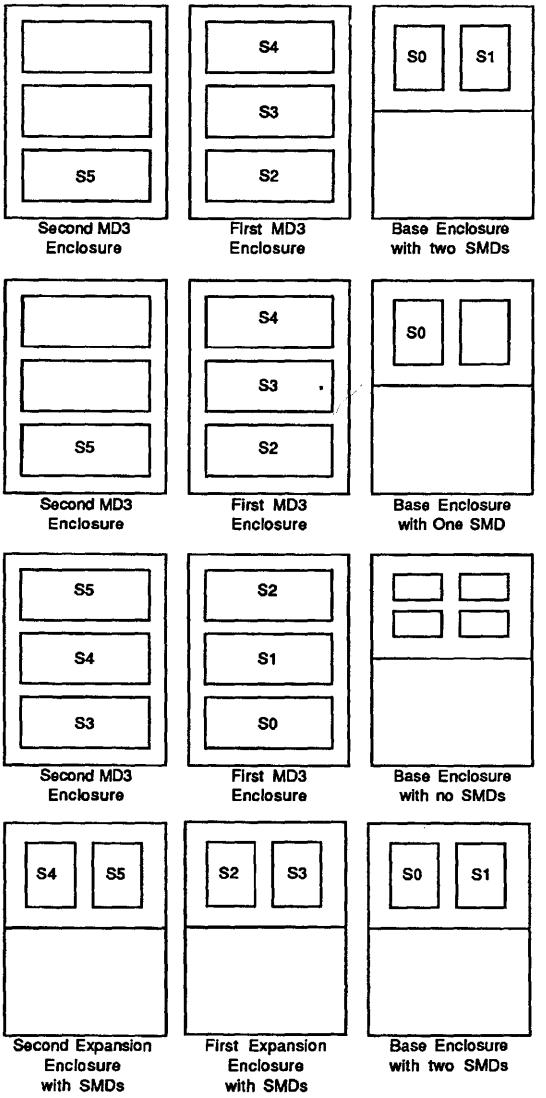


Figure B–3 **Processor Board Numbering Scheme**

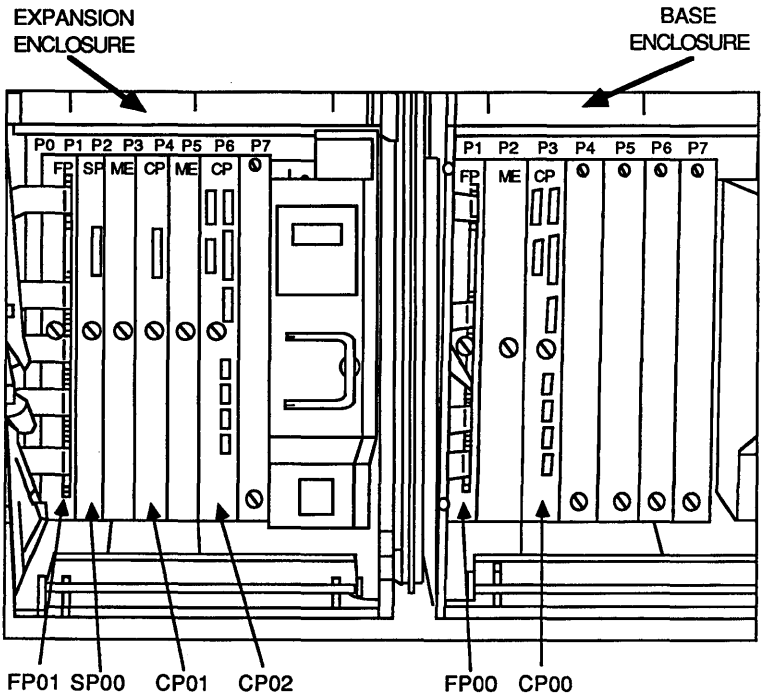


Table B–1 **Processor Slot Numbering Scheme**

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
Expansion 4	30	31	32	33	34	35	36	37
Expansion 5	20	21	22	23	24	25	26	27

Glossary

AdminAgent. AdminAgent is the system service that manages the execution of the master utilities on XE520 processors. The AdminAgent can be run in either single mode or multiple mode.

archive. The archive procedure is used to backup system files onto portable media.

archive volume. An archive volume is a portable medium that is used to archive (back up) files.

ASCII (American National Standard Code for Information Interchange). ASCII is a 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. Once started up, a background process runs with no interaction with the user through the terminal. Another process is free to use the terminal.

backup. See **archive**.

bad spot. A bad spot is any part of a hard disk drive that is faulty and cannot record information.

Batch utility. A batch utility provides a job stream capability and allows programs to run in a processor's secondary partitions while a user–interactive program runs in a primary partition.

block. A block is a 512–byte subdivision of data on the disk. A block is also referred to as a sector.

bootup. Bootup is the operation that starts up the system when the keyswitch on the XE520 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.

BTOS. BTOS is the Unisys workstation operating system. All XE520 boards run a version of BTOS.

BTOS Executive. See **Executive**.

BTOS status codes. BTOS status codes are 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 BTOS workstation refers to a Unisys model B21, B22, B26, B27, or B28 workstation.

built–in disk device. A built–in disk device is 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 XE520 or MD3 enclosure and used for the mass storage of programs and data.

central processing unit (CPU). The CPU is the part of the computer, that performs most of the fundamental processing. In the XE520 system, this term is sometimes used interchangeably with the term processor.

CLI. See **Command Line Interpreter**.

cluster line. Cluster line refers to the RS–422 cluster communications port connection which allows workstations to communicate with the XE520.

Cluster Processor (CP). A cluster processor is a board in the XE520 system that runs communications software and supports 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 command file is 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 command form is 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). The command line interpreter is an interface program that interprets statements and executes BTOS run files on XE520 processors.

CP. See **Cluster Processor**.

CPU. See **central processing unit.**

crash dump. A crash dump is a file that contains a copy of a processor's memory image at the time of a nonrecoverable failure (crash). Such a file can be used by Unisys personnel to analyze the cause of a processor failure.

CRC. See **Cyclical Redundancy Check.**

customized mode. The customized mode is the state of the system when it is booted up using a customized set of system configuration files.

Customizer. In BTOS, customizer is 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 cyclical redundancy check is a method of data error detection in which a two–byte block check character 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 cylinder is a set of tracks on a storage device that can be accessed together as a unit.

DAM. See **Direct Access Method.**

data volume. A data volume is used for data and program storage.

debugger. A debugger is 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.

Demand paging is a form of memory management that maintayns onboard memory for only those parts of the program code and data required for execution.

device. A device is a physical hardware entity, such as a terminal, printer, tape drive, or other input/output unit.

Direct Access Method (DAM). The direct access method is a file access method providing random and non–overlapped input or output of records.

directory. A directory is a related group of files on a volume. Directories are the second level of the BTOS three–level hiearchical file system.

disk cartridge. A disk cartridge is a magnetic disk storage medium utilizing a hard disk enclosed in a portable cartridge. Disk cartridges are used with a disk cartridge drive of an XE520 base enclosure.

disk device. A disk device is a hardware component that is used to store data. An XE520 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. The disk extent is one or more contiguous disk sectors that contain all or part of a file.

disk initialization. See **volume initialization.**

Disk Processor (DP). The disk processor is a board in an XE520 system that is formed by connecting SC to SP. The DP supports I/O to half–inch magnetic tape drives and SMD disks.

DP. See **Disk Processor.**

ECC. See **Error Correction Code.**

Error Correction Code (ECC). The error correction code is 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.

Executive. The Executive is an interactive BTOS application program through which the various BTOS utilities can be executed through command forms.

File Processor (FP). The file processor is a board in an XE520 system that supports I/O operations to disk devices.

FP. See **File Processor**.

file system. A file system is a collection of files that are stored on the same logical disk device. A BTOS file system is made up of a three–level hierarchy that includes a volume, directories, and the files contained within each directory.

hexadecimal values. Hexadecimal values are used in the notation system of representing numbers in base 16.

home directory. Home directory is 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). The indexed sequential access method is a programming tool that provides sequencing of file records on disk and direct access to those records using an index.

installation logs. Installation logs are logs that provide a record of software installation operations.

Inter–CPU Communication (ICC). Inter–CPU Communication is the Intercommunications protocol used by XE520 processors when communicating across the system bus.

I/O. Input/output.

I/O device configuration files. I/O device configuration files 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. ISAM server is the application that manages the execution of ISAM–related operations on a BTOS processor.

JCL. See **Job Control Language**.

Job Control Language (JCL). Job control language are statements submitted to CLI for batch execution.

kb. See kilobit.

kB. See kilobyte.

kilobit. A kilobit is 1024 bits.

kilobyte. A kilobyte is 1024 bytes.

language development software. Language development software are the software tools that support BTOS programming languages.

log file. A log file is a file in BTOS, which contains a record of operations performed by BTOS processors.

master commands. Master commands are a set of BTOS commands, executed from the Executive, which invoke the corresponding utilities that run on an XE520 processor. See utilities.

master configuration file. The master configuration file is 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 master processor is the first processor in the XE520 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.

Mb. See megabyte.

mBTOS CONFIG utility. mBTOS config utility is 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. A megabyte is 1,024,000 bytes.

Memory Expansion (ME) Board. A memory expansion board is attached to a processor board to supply additional processor memory capacity.

menu. In BTOS, a menu is a terminal screen that provides a listing of operations from which a user selects to execute the desired operation.

Multidisk 3 (MD3). A Multidisk 3 is a freestanding enclosure supporting up to three 135 Mb (formatted) SMD disk drives.

normal mode. Normal mode is an XE520 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 null string is a sequence of blank characters that usually does not change the default condition set up in the original BTOS command or submit file.

operating mode. The XE520's operating mode is determined by any one of four sets of system configuration files and BTOS processor versions used to boot up the XE520. The particular set of system configuration files used during boot–up time depends on the keyswitch position.

operating system. The operating system is 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. A partition is an assigned area of the XE520 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. See **primary partition** and **secondary partition**.

password. A password is a string of characters used as part 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. The path name is the full description that identifies a file's position in the file system. It includes a file's volume name, directory name and file name.

port. The port is the part of a processor dedicated to a single data channel for receiving data from, or transmitting data to, one or more external remote devices.

primary partition. A primary partition is an assigned area of an XE520 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 printer spooler is 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 printer spooler manager is a system service that manages the transfer of data from disk files to printers.

printer translation file. In BTOS, the printer translation file 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. The processor configuration file is a file that defines the hardware and software configuration associated with a processor board.

processor initialization file. The processor initialization file is a file that defines the system services that are to be run on a processor when the system is started up.

product content file. The product content file is 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. The protection level is 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 server. QIC server is an application that controls the execution of I/O operations to a QIC tape drive.

QIC tape. See **quarter–inch cartridge tape**.

quarter–inch cartridge (QIC) tape. Quarter–inch cartridge tape is a magnetic tape storage medium that utilizes quarter–inch–wide tape contained in a hard plastic cartridge.

queue index file. Queue index file is a file that defines the queues for which the Queue Manager is responsible.

Queue Manager. Queue manager is 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). Random–access memory is the onboard processor memory to which and from which data can be written and read nonsequentially.

read–only memory (ROM). Read–only memory is 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. Register is a temporary memory location for data.

release medium. In BTOS, the release medium is a disk cartridge, QIC tape, or half–inch tape on which Unisys release software is stored.

restore. Restore is the procedure by which archive files are written from an archive medium to an XE520 volume.

restricted mode. The restricted mode is an XE520 operating mode that is determined by the keyswitch setting of the XE520 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**.

RS–232–C. An RS–232–C designation refers to an industry specification developed to standardize the interface between different types of communications equipment.

RS–422. An RS–422 is a high–speed communications standard used in an XE520 system to link cluster workstations.

run statement. A run statement is a line of text that can be included in a JCL file for execution on a processor by the CLI.

sector. A sector is the smallest addressable portion of a track or band on a hard or floppy disk. See **block.**

secondary partition. A secondary partition is an assigned area of an XE520 processor's memory in which applications, including some system services, that do not require user interaction can be run.

signon file. A signon file is a file required for each user that enables the user to log into the system. The file also 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.

SMD. See **Storage Module Device**.

SP. See **Storage Processor**.

special file. See **device file**.

spooler. See **printer spooler.**

spooler configuration file. A spooler configuration file is 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 status display is the display on the XE520 base enclosure's front panel through which the status of software installation, system start–up operations, and error conditions are reported.

Storage Controller (SC). The storage controller is a board used with SP to form DP, which controls SMDs.

Storage Module Device (SMD). The storage module device is a 133 Mb (formatted) Memorex 166 disk drive.

Storage Processor (SP). The storage processor is a processor board in the XE520 system that 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 submit file causes a series of BTOS utilities to be executed.

surface test. A surface test is 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.

system bus. The system bus is the path over which the system processors communicate.

system configuration files. System configuration files are text files that define the XE520 BTOS hardware and software configuration.

system crash status words. System crash status words are 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 system disk is the disk on which the XE520 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 system image is the run file for an operating system.

system log. The system log is a log, displayable through the PLOG command, that contains a detailed account of system status, errors, and failures.

system volume. The system volume is a volume from which you can boot the XE520.

terminal. A terminal is a device, usually equipped with a keyboard and a display, which is capable of sending and receiving information over a communication channel.

tape server. The tape server is an application that controls the execution of I/O operations to a half–inch tape drive.

Temporary Directory Filter. A temporary directory filter is a system service in XE520 BTOS that causes all temporary files, regardless of what application created them, to be stored in one directory on an XE520 volume.

Terminal Processor (TP). The terminal processor is a processor board in an XE520 system that supports a parallel printer and up to ten RS–232–C serial devices.

TP. See **Terminal Processor**.

user command file. A user command file is a file in BTOS 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, volume is the complete file system unit of information stored on a formatted disk.

volume fragmentation. Volume fragmentation is 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). Volume home block is 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. Volume initialization is the procedure performed by the mIVOLUME utility in which an XE520 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.

wildcard character. The BTOS system allows two wildcard 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 wildcard characters in file specifications. The system then tries to match the portion of the name that appears before or after the wildcard character and performs the requested operation for each matched file name.

x, y, z blocks. X, y, z blocks are operating system parameters that control the number and size of cluster communication blocks.

Index

C

H

I

O

P

Q

R

S

T

U

V

W

X

NOTES

NOTES

NOTES

Help Us To Help You

Publication Title

Form Number Date

Unisys 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

Help Us To Help You

Publication Title

Form Number Date

Unisys 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

No Postage
necessary
if mailed in the
United States

BUSINESS REPLY MAIL

First Class Permit No. 817 Detroit, MI 48232

Postage Will Be Paid By Addressee

Unisys Corporation
ATTN: Corporate Product Information
P.O. Box 418
Detroit, MI 48232-9975 USA

No Postage
necessary
if mailed in the
United States

BUSINESS REPLY MAIL

First Class Permit No. 817 Detroit, MI 48232

Postage Will Be Paid By Addressee

Unisys Corporation
ATTN: Corporate Product Information
P.O. Box 418
Detroit, MI 48232-9975 USA