



**CTOS**

**System  
Administration  
Guide**

**UNISYS**

**UNISYS**

**CTOS<sup>®</sup>**

**System  
Administration  
Guide**

Copyright © 1990, 1991, 1992, 1993 Unisys Corporation  
All Rights Reserved  
Unisys is a trademark of Unisys Corporation

CTOS III 1.1  
CTOS II 3.4.4  
CTOS I 3.4  
CTOS/XE 3.4.100  
Standard Software 12.3  
System Build Utilities 1.1/3.4  
Video Access Method 4.2  
Priced Item

October 1993

Printed in USA  
4357 4599-300



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 THIS DOCUMENT.** Any product and related material disclosed herein are only furnished pursuant and subject to the terms and conditions of a duly executed Program Product License or Agreement to purchase or lease equipment. The only warranties made by Unisys, if any, with respect to the products described in this document are set forth in such License or Agreement. Unisys cannot accept any financial or other responsibility that may be the result of your use of the information or software material, including direct, indirect, special or consequential damages.

You should be 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.

**RESTRICTED RIGHTS LEGEND.** Use, reproduction, or disclosure is subject to the restrictions set forth in DFARS 252.227-7013 and FAR 52.227-14 for commercial computer software.

CTOS, NGEN, and SuperGen are registered trademarks of Convergent Technologies, Inc.

Context Manager, CTOS/Vpc, Document Designer, Genenc Print System, Image Designer, Print Manager, Series 286i, Series 386i, shared resource processor, Solution Designer, SRP, Voice Processor, X-Bus, and X-Bus+ are trademarks of Convergent Technologies, Inc.

OFIS is a registered trademark of Unisys Corporation.

AT, IBM, and IBM PC are registered trademarks of International Business Machines Corporation. Intel is a registered trademark of Intel Corporation. MS-DOS and Microsoft are registered trademarks of Microsoft Corporation.

# Contents

## About This Guide

<b>Introduction</b> .....	xxiii
<b>What This Guide Covers</b> .....	xxiii
<b>Who Should Use This Guide</b> .....	xxiv
<b>How This Guide Is Organized</b> .....	xxiv
<b>What Is New in This Guide</b> .....	xxvii
<b>Terminology Used in This Guide</b> .....	xxvii
<b>Conventions Used in This Guide</b> .....	xxviii
<b>Where to Find More Information</b> .....	xxix
Hardware Installation Guides .....	xxix
Administration Guides for Applications .....	xxx
Software Development Manuals .....	xxx

## Section 1. If You Are New to System Administration

<b>Overview</b> .....	1-1
<b>Terms</b> .....	1-1
<b>What Does “Configuration” Mean?</b> .....	1-3
<b>Administrative Duties</b> .....	1-3
Setting Up a System .....	1-4
Other Administrative Tasks .....	1-5

## Section 2. Understanding Hardware

<b>What Is a Cluster?</b> .....	2-1
<b>Workstation Hardware</b> .....	2-2
Workstation Models .....	2-2
X-Bus Workstations .....	2-2
X-Bus+ Workstations .....	2-3
EISA and ISA Workstations .....	2-3
Diskless Workstations .....	2-3
Processors .....	2-3
CTOS I/O Processor Cards .....	2-4
Processor Identifiers .....	2-4

## Contents

---

Cartridges, Modules, and Expansion Cards . . . . .	2-5
Monitors . . . . .	2-5
Keyboards . . . . .	2-6
<b>Shared Resource Processor Hardware . . . . .</b>	<b>2-6</b>
Cabinets . . . . .	2-6
Processor Boards . . . . .	2-8
Protected-Mode Processors . . . . .	2-8
Real-Mode Processors . . . . .	2-9
The Master Processor . . . . .	2-9
Processor Identifiers . . . . .	2-10
Protected-Mode Processor Numbers . . . . .	2-11
Real-Mode Processor Numbers . . . . .	2-11
Memory Expansion Boards . . . . .	2-11
<b>Understanding EISA/ISA Workstations . . . . .</b>	<b>2-13</b>
Industry Standard Hardware . . . . .	2-13
EISA and ISA Cards . . . . .	2-13
EISA and ISA Workstation Models . . . . .	2-14
What Is the BIOS? . . . . .	2-14
What Is CMOS? . . . . .	2-14
Hardware Configuration Programs . . . . .	2-15
Setup . . . . .	2-15
System Configuration Utility . . . . .	2-15
Disks . . . . .	2-16
SCSI Drives . . . . .	2-16
IDE Drives . . . . .	2-16
Cards . . . . .	2-16
Cards for ISA Workstations . . . . .	2-16
Cards for EISA Workstations . . . . .	2-17
Video . . . . .	2-17
CTOS VideoCard . . . . .	2-17
Standard VGA Display . . . . .	2-18
Keyboards and Mouse Devices . . . . .	2-18

### **Section 3. Understanding System Software**

<b>What Is System Software? . . . . .</b>	<b>3-1</b>
<b>Standard Software . . . . .</b>	<b>3-1</b>
<b>Video Access Method . . . . .</b>	<b>3-1</b>
<b>Workstation Operating Systems . . . . .</b>	<b>3-2</b>
Server Workstations . . . . .	3-4
Client Workstations With Local File Systems . . . . .	3-4
Diskless Client Workstations . . . . .	3-5
<b>SRP Operating Systems . . . . .</b>	<b>3-5</b>



**Section 4. Using Administrative Tools**

<b>What Tools Are Available?</b> .....	4-1
<b>System Manager</b> .....	4-1
Starting the System Manager .....	4-2
Using the System Manager on a Workstation .....	4-3
With the Keyboard .....	4-3
With the Mouse .....	4-3
Function Keys .....	4-4
Using the System Manager on an SRP .....	4-4
Starting the System Manager With	
Cluster View .....	4-4
System Manager Display on the SRP .....	4-5
<b>The Editor</b> .....	4-6
Starting and Exiting the Editor .....	4-6
Editing Files .....	4-6
Opening, Closing, and Saving Files .....	4-7
Searching for Text .....	4-8
Replacing Text .....	4-8
Manipulating Files .....	4-8
<b>Cluster View</b> .....	4-10
Using Cluster View on a Shared Resource	
Processor .....	4-10
Using Cluster View on a Workstation Server .....	4-10
Cluster View System Services .....	4-11
Remote Keyboard Video Service .....	4-11
Remote User Manager .....	4-11
Cluster View Commands .....	4-12
Installing the Cluster View Services .....	4-13
Starting Cluster View .....	4-13
Working in a Cluster View Session .....	4-15
Displaying the Cluster View Menu .....	4-15
Exiting Cluster View .....	4-16
Using Cluster View on a Single-Processor	
Workstation .....	4-16
Installing RKVS, the Login Service, and	
RUM .....	4-17
Using the Keyboard and Monitor .....	4-17

## Section 5. Bootstrapping

<b>How a System Bootstraps</b> .....	5-1
<b>Bootstrapping a Workstation</b> .....	5-1
Bootstrapping I/O Processors .....	5-2
Bootstrapping a Workstation From a Server .....	5-3
Workstation-Type Numbers .....	5-3
Using the Bootstrap Menu .....	5-5
Procedure for X-Bus and X-Bus+ Workstations ..	5-6
Procedure for EISA/ISA Workstations .....	5-6
Indirect Bootstrapping .....	5-7
Workstation Hardware IDs .....	5-7
Using the Bootstrap Command .....	5-8
<b>Bootstrapping a Shared Resource Processor</b> .....	5-8
Keyswitch Positions .....	5-11
Keyswitch-Controlled Files .....	5-11

## Section 6. Implementing System Security

<b>How Passwords Work</b> .....	6-1
<b>Protecting Volumes</b> .....	6-1
Assigning a Password to the System Volume .....	6-2
Assigning Passwords to Other Volumes .....	6-2
Changing a Volume Password .....	6-4
<b>Protecting Directories</b> .....	6-5
Assigning a Password to a Directory .....	6-7
Changing a Directory Password .....	6-8
Protecting the <Sys> Directory .....	6-9
Limiting Access to Directories .....	6-9
<b>Protecting Files</b> .....	6-9
Assigning a Protection Level to a Group of Files .....	6-10
Assigning a Unique Password to a File .....	6-11
<b>Restricting Access to the System</b> .....	6-11
Allowing Access to a Single Directory .....	6-12
Eliminating Known User Names .....	6-12
Setting Up a Videoless Workstation Server .....	6-13
<b>Limiting Access to Commands</b> .....	6-14
Installing the Command Access Service .....	6-14
Configuring the Command Access Service .....	6-15

Creating the Command Access Service Configuration File .....	6-15
Protecting the Command Access Service Configuration File .....	6-16
Using the Command Access Service Log File .....	6-16
Allowing Access to Users on Other Nodes .....	6-17

## Section 7. Customizing User Environments

<b>What Is a User Configuration File?</b> .....	7-1
<b>Creating a User File</b> .....	7-1
<b>Modifying a User File</b> .....	7-3
<b>Editing a User File Manually</b> .....	7-3
File Specifications for User Files .....	7-4
User File Format .....	7-4
<b>Adding a User File Option</b> .....	7-5
<b>Creating a Working Environment</b> .....	7-6
Defining the Default Environment .....	7-6
Defining Optional Environments .....	7-8
Keywords Recognized by the SignOn Program .....	7-8
The SignOn Chain File Entry .....	7-10
The SignOn Exit File Entry .....	7-10
<b>Limiting Access to the System</b> .....	7-10
Removing the Default User File .....	7-11
Assigning Passwords to User Names .....	7-11
<b>Signing On Automatically</b> .....	7-12
<b>Signing On With a Magnetic Card Reader</b> .....	7-13
<b>User File Options for Standard Software</b> .....	7-14
SignOn Options .....	7-14
Executive Option .....	7-16
Mouse Options .....	7-16
Installation Manager Options .....	7-17
Cluster View Options .....	7-20
Editor Option .....	7-22

## Section 8. Installing Applications

<b>Software Packages</b> .....	8-1
<b>Planning the Installation</b> .....	8-1
<b>What Applications Are Available?</b> .....	8-2
Office Automation Applications .....	8-2
Communications Applications .....	8-2
Other Applications .....	8-3



<b>Using the Installation Manager</b> .....	8-3
<b>Installing Applications</b> .....	8-4
<b>Installation Manager Features</b> .....	8-10
Installing From Floppy Disks .....	8-10
Installing From QIC Tape .....	8-10
Installing From Compact Disc .....	8-10
Installing From the Server .....	8-10
Installing Public Software .....	8-11
Using the Log File .....	8-11
Removing an Application .....	8-12
<b>Recovering From Installation Failures</b> .....	8-12
<b>Restarting an Installation</b> .....	8-13
<b>Loadable Requests</b> .....	8-13
<b>Common Installation Problems</b> .....	8-14

## Section 9. Installing System Services

<b>What Is a System Service?</b> .....	9-1
<b>What System Services Do You Need?</b> .....	9-2
Standard Software System Services .....	9-2
Generic Print System Services .....	9-4
Electronic Mail Services .....	9-6
Network System Services .....	9-8
Other System Services .....	9-8
<b>Where to Install System Services</b> .....	9-9
<b>Calculating Memory Requirements</b> .....	9-11
Operating Systems Without Virtual Memory .....	9-11
Virtual Memory Operating Systems .....	9-11
How Much Memory Is Available? .....	9-11
Starting Partition Status on a Single-Processor Workstation .....	9-13
Starting Partition Status on a Multiprocessor System .....	9-13
How Much Memory Is Required? .....	9-13
<b>Installing System Services on a Single-Processor Workstation</b> .....	9-14
Installing From the Executive .....	9-15
Installing During System Initialization .....	9-16
Creating a System Initialization File .....	9-17
Workstations That Boot From the Server .....	9-17

<b>Installing System Services on a Multiprocessor Workstation</b> .....	9-19
Installing With Cluster View .....	9-19
Installing During System Initialization .....	9-20
JCL Statements for Multiprocessor Systems .....	9-22
Specifying a Processor .....	9-22
<b>Installing System Services on an SRP</b> .....	9-22
Specifying a Processor .....	9-24
Using Keyswitch Files .....	9-24
<b>JCL Syntax</b> .....	9-24
Specifying Run Files and Command Names .....	9-25
Specifying Parameters .....	9-25
Specifying Subparameters .....	9-26
Entering Multiple-Line Parameters .....	9-26
Entering Comments .....	9-26
<b>Installing the Remote User Manager</b> .....	9-27
<b>Using the RunNoWait JCL Statement</b> .....	9-27
<b>Common Problems With System Services</b> .....	9-29

## Section 10. Accessing Data Throughout the Cluster

<b>Using Disks on the Server</b> .....	10-1
<b>Accessing Disks on Client Workstations</b> .....	10-1
Configuring Cluster File Access .....	10-2
Installing the CFA Server Service .....	10-3
Installing the CFA Workstation Agent .....	10-3
Installing the CFA File Filter .....	10-4
Using Cluster File Access .....	10-4
Accessing Floppy Disk Drives .....	10-5
Displaying CFA Volume Names .....	10-5
<b>Accessing Additional Resources</b> .....	10-5

## Section 11. Adding Hard Disks

<b>What Is a CTOS Volume?</b> .....	11-1
<b>Workstation Disks</b> .....	11-1
Workstation Disk Compatibility .....	11-2
Device Names for Workstation Hard Disks .....	11-2
Disk Type and Bad Spot Report .....	11-3
Partitioned Disks .....	11-3
<b>SRP Disks</b> .....	11-4
SRP Disk Compatibility .....	11-4
Device Names for SRP Disk Drives .....	11-4

## Contents

---

<b>Initializing an Unformatted Disk</b> .....	11-5
<b>Creating a Bad Spots File</b> .....	11-10
Bad Spot Formats for Non-SCSI Disks .....	11-10
Bad Spot Format for SCSI Disks .....	11-11
<b>Reinitializing a Valid Volume</b> .....	11-12
<b>Using Parameter Templates</b> .....	11-13
Configuration File Format .....	11-13
Using Format Templates .....	11-14
Adding a Format Template .....	11-17
Format Template Parameters .....	11-18
Using Device Templates .....	11-23
Adding a Device Template .....	11-24
Device Template Parameters .....	11-25
<b>Using the Configure Disk Command</b> .....	11-28
<b>Formatting a Partitioned Disk</b> .....	11-29
Activating a non-CTOS Partition .....	11-29
Reactivating a CTOS Partition .....	19-30
<b>Correcting Input/Output (I/O) Errors</b> .....	19-30
Specifying Bad Spots .....	11-31
Running Surface Tests .....	11-31
Reinitializing the Disk .....	11-32
<b>Reinitializing Corrupted Volumes</b> .....	11-33
<b>Optimizing Disk Space</b> .....	11-33

## Section 12. Using Tape and CD-ROM Drives

<b>Tape Drives</b> .....	12-1
What Kind of Tapes to Use .....	12-2
Hardware and Software Requirements .....	12-2
Installing the Sequential Access Service .....	12-3
Configuring a Tape Drive .....	12-4
Preparing Tapes for Use .....	12-5
Write-Enabling Half-Inch Tapes .....	12-5
<b>CD-ROM Drives</b> .....	12-6
Hardware and Software Requirements .....	12-6
Installing the CD-ROM Service .....	12-7



**Section 13. Backing Up and Restoring Data**

<b>Performing Routine Backups</b> .....	13-1
<b>Cleaning Up Disks Before Backups</b> .....	13-1
<b>Performing a Complete Volume Backup</b> .....	13-2
<b>Performing an Incremental Backup</b> .....	13-6
<b>Performing Backups With Cluster View</b> .....	13-7
<b>Restoring Backups</b> .....	13-8
Restoring a Complete Backup .....	13-9
Restoring Portions of an Archive Dataset .....	13-12
<b>Recovering a Corrupted Volume</b> .....	13-12
Identifying a Corrupted Volume .....	13-13
Backing Up a Corrupted Volume .....	13-14
Troubleshooting Disk Problems .....	13-14
Restoring Data .....	13-15

**Section 14. Using a File-System Cache**

<b>What Is a File-System Cache?</b> .....	14-1
<b>How Caching Works</b> .....	14-1
Write-Through Caching .....	14-3
Write-Behind Caching .....	14-3
<b>Configuring Cache Memory</b> .....	14-3
<b>Setting File Attributes for Caching</b> .....	14-4
Disabling Files for Caching .....	14-4
Enabling Files for Caching .....	14-5
<b>Using the Cache as a RAM Disk</b> .....	14-5
<b>Caching From Client Workstation to Server</b> .....	14-6
Caching on Cluster IOPs .....	14-7
Caching on Diskless Workstations .....	14-7
<b>Remote Caching on an SRP</b> .....	14-7
Sharing a Cache .....	14-8
Configuring a Remote Cache .....	14-9

**Section 15. Optimizing System Performance**

<b>Optimizing Partitioned Memory</b> .....	15-1
Using Context Manager .....	15-1
Calculating Partition Sizes .....	15-2
Using Static Partitions .....	15-3
Using Variable Partitions .....	15-3
Code Sharing .....	15-4

<b>Optimizing Virtual Memory</b> .....	15-4
Suppressing Global Policy .....	15-5
Correcting Global Thrashing .....	15-5
<b>Allocating Queues</b> .....	15-6
Dynamic Queues .....	15-6
Static Queues .....	15-7
<b>Optimizing Use of Disk Space</b> .....	15-7
Moving the "Scratch" Volume .....	15-8
Moving Applications .....	15-8
<b>Conserving Memory on Single-Processor CTOS III Servers</b> .....	15-9
<b>Optimizing Memory Usage on an SRP</b> .....	15-9
Isolating Disk-Intensive Applications .....	15-9
Moving Communications Services .....	15-10
Avoiding Interprocessor Data Transfers .....	15-10
<b>Adjusting Memory Blocks</b> .....	15-10
What Are Blocks? .....	15-11
X-Blocks .....	15-11
W-, Y-, and Z-Blocks .....	15-12
<b>Using a Cache Memory Disk</b> .....	15-12

## Section 16. Configuring Operating Systems

<b>The Operating System Configuration File</b> .....	16-1
Configuration File Format .....	16-1
Configuration File Syntax .....	16-2
<b>Configuration Files for Single-Processor Workstations</b> ..	16-3
<b>Configuration File for Multiprocessor Systems</b> .....	16-4
Boot Section .....	16-6
Processor Section .....	16-6
Processor Defaults .....	16-7
Entries for Specific Processors .....	16-7
<b>Editing the Configuration File</b> .....	16-8
<b>Creating WsNNN&gt;Config.sys</b> .....	16-8
<b>Using Special-Purpose Configuration Files</b> .....	16-9
Specifying a Configuration File With the Bootstrap Command .....	16-9
Associating Configuration Files With SRP Keyswitch Positions .....	16-9
<b>Configurable Parameters</b> .....	16-10

**Section 17. Building a Customized Operating System**

<b>When to Customize an Operating System</b> .....	17-1
<b>Installing the System Build Utilities</b> .....	17-2
<b>Using the Mini-Customizer</b> .....	17-2
<b>Building an Operating System</b> .....	17-3
Installing the Development Utilities .....	17-3
Making Changes to the Source Code .....	17-3
Assembling and Linking .....	17-7
File System Prefix File .....	17-7
Cluster Agent Prefix File .....	17-8
Operating System Prefix File .....	17-9
<b>Testing the New Operating System</b> .....	17-10
On a Workstation .....	17-10
On an SRP Master Processor .....	17-11
On Other SRP Processors .....	17-12
<b>Installing the New Operating System</b> .....	17-12
<b>Troubleshooting SysGen Errors</b> .....	17-13
Assembly Errors .....	17-13
Link Errors .....	17-13
Bootstrap Errors .....	17-14

**Section 18. Customizing Standard Software**

<b>What You Can Customize</b> .....	18-1
<b>Customizing Message Files</b> .....	18-1
Generating a Message Text File .....	18-2
Editing a Message Text File .....	18-2
Creating a Binary Message File .....	18-3
Merging Message Files .....	18-4
<b>Customizing Templates</b> .....	18-4
<b>Customizing the SignOn Screen</b> .....	18-5
Activating the Window-Interface SignOn Screen .....	18-6
Disabling the Window-Interface SignOn Screen .....	18-7
Editing the SignOn Heading .....	18-7
Editing the SignOn Text File .....	18-7
<b>Setting Color Attributes</b> .....	18-8
Displaying Color on the SignOn Screen .....	18-8
Mixing Color and Other Video Attributes .....	18-9



## Section 19. Troubleshooting

<b>Diagnosing Problems</b> .....	19-1
PLog .....	19-1
Example 1: System Bootstrap Event .....	19-2
Example 2: Disk Error .....	19-2
Example 3: System Initialization Error .....	19-3
Example 4: Processor Crash .....	19-3
Cluster Status .....	19-4
Partition Status .....	19-5
<b>Workstation Troubleshooting</b> .....	19-5
Workstation Power Does Not Come On .....	19-6
Workstation Does Not Bootstrap .....	19-6
Module or Card Is Not Recognized .....	19-7
Keyboard Does Not Work .....	19-10
Video Does Not Work Properly .....	19-11
Workstation Does Not Communicate With the Server .....	19-12
Application Cannot Be Started .....	19-13
Application Is Running Slowly .....	19-14
<b>SRP Troubleshooting</b> .....	19-15
Hardware Installation Problems .....	19-15
Processor Crashes .....	19-15
Interpreting the Front Panel Display .....	19-16
Interpreting Processor LEDs .....	19-16
Basic Procedures for SRP Processor Troubleshooting .....	19-21
SRP Processor Crashes During Bootstrapping .....	19-22
Isolating Hardware Problems .....	19-23
Isolating Software Problems .....	19-24
Errors During System Service Installation .....	19-24
Intermittent System Crashes .....	19-25
<b>Collecting a Crash Dump</b> .....	19-25
Performing Crash Dumps on a Workstation .....	19-26
X-Bus Workstations .....	19-26
X-Bus+ Workstations .....	19-27
Diskless Workstations .....	19-27
ISA Workstations and Single-Processor EISA Workstations .....	19-27
Multiprocessor EISA Workstations .....	19-28
Creating an Extended Crash Dump File .....	19-28

Suppressing Crash Dumping .....	19-29
Performing Crash Dumps on an SRP .....	19-29
Master Processor .....	19-30
Other SRP Processors .....	19-30
<b>What If a System Will Not Bootstrap?</b> .....	19-30
Bootstrapping From the Server .....	19-31
Bootstrapping From a Floppy Disk .....	19-32
Regular and High-Capacity Drives .....	19-32
Creating a Message File .....	19-33
Creating a Bootable Floppy Disk Set .....	19-34
Labeling the Bootable Floppy Disk Set .....	19-36
Using the Bootable Floppy Disk Set .....	19-36
Bootstrapping From QIC Tape .....	19-37
Using the Bootable Tape .....	19-37
Creating the Bootable Tape .....	19-38
Error 21 on the Front Panel .....	19-39
<b>Converting Hexadecimal Error Codes</b> .....	19-39

## Appendix A. Workstation Bootstrap and Initialization Codes

Introduction .....	A-1
<b>Operating System Initialization Status Codes</b> .....	A-3
Initialization Codes .....	A-3
<b>Boot ROM Codes</b> .....	A-5
Boot ROM Codes for X-Bus Workstations and the ClientCard .....	A-7
Boot ROM Codes for B24 Workstations .....	A-40
2681 Error Codes .....	A-45
Keyboard Test Error Codes .....	A-46
ROM Checksum Test Error Codes .....	A-47
RTC Test Error Codes .....	A-48
Interrupt Test Error Codes .....	A-48
VSDD Error Codes .....	A-49
7201 MPSC Error Codes .....	A-49
Optional 2681 .....	A-50
Communications Error Codes .....	A-50
Security Module Test Error Codes .....	A-51
NVM Test Error Codes .....	A-53
RAM Test Error Code .....	A-53
Boot Errors for EISA and ISA Workstations .....	A-55
Boot ROM Codes for SuperGen Series 5000 Workstations .....	A-56
Boot ROM Codes for SG2100-320 Workstations .....	A-74

**Appendix B. Shared Resource Processor Front Panel and Processor Status Codes**

<b>Front Panel Status Display</b> . . . . .	B-1
<b>General Processor LED Display</b> . . . . .	B-2
<b>186 Processor Board LED Display</b> . . . . .	B-2
<b>Front Panel Status Codes</b> . . . . .	B-3
Boot Medium Error Codes . . . . .	B-4
Disk Hardware Error Codes . . . . .	B-4
Processor Error Codes . . . . .	B-5
Software Error Codes . . . . .	B-5
Expansion Enclosure Error Codes . . . . .	B-6
Disk Hardware Error Codes . . . . .	B-6
<b>186-Board Processor LED Codes</b> . . . . .	B-6
Processor LED Codes While Front Panel Displays 01 . . . . .	B-7
Processor LED Codes While Front Panel Displays 21 . . . . .	B-7
Processor LED Codes While Front Panel Displays 39 . . . . .	B-8
General Status Codes . . . . .	B-9
<b>386-Board Processor LED Codes</b> . . . . .	B-9
Processor LED Codes While Front Panel Displays 01 . . . . .	B-10
Processor LED Codes While Front Panel Displays 39 . . . . .	B-11
Processor LED Codes During Booting and Crash-Dumping . . . . .	B-12
<b>System Crash Status Words</b> . . . . .	B-12

**Appendix C. Configuring Standard VGA**

<b>Customizing the Display for Character-Based Applications</b> .....	C-1
Making Simple Modifications With Executive Commands .....	C-2
Changing Color Mapping .....	C-2
Mapping Struck-Through to a Color .....	C-3
Substituting Bold for Underlining .....	C-3
Displaying Current Mapping Attributes .....	C-4
<b>Making More Extensive Modifications</b> .....	C-5
Attribute Mapping Theory .....	C-5
The VGA Character Attribute Mapper .....	C-8

**Appendix D. CTOS I/O Processor Card Error Codes**

Introduction .....	D-1
Error Codes .....	D-2

<b>Glossary</b>	Glossary-1
-----------------	------------

<b>Index</b>	Index-1
--------------	---------





# Figures

2-1.	Simple Cluster Hardware Configuration . . . . .	2-1
2-2.	SRP Primary Cabinet . . . . .	2-7
2-3.	SRP Master Processor . . . . .	2-10
2-4.	SRP Processor Identifiers . . . . .	2-12
3-1.	Operating System Identification . . . . .	3-3
4-1.	System Manager Display on a Workstation . . . . .	4-2
4-2.	System Manager Display on an SRP . . . . .	4-5
4-3.	Cluster View Menu . . . . .	4-15
5-1.	Bootstrap Menu . . . . .	5-5
5-2.	SRP Bootstrap Sequence . . . . .	5-9
6-1.	Command Access Service Configuration File . . . . .	6-15
7-1.	User Configuration File . . . . .	7-4
7-2.	User Configuration File, Multiple Environments . . . . .	7-7
9-1.	GPS System Services in a Cluster . . . . .	9-5
9-2.	Electronic Mail System Services in a Cluster . . . . .	9-7
9-3.	System Services Installed Throughout a Cluster . . . . .	9-10
9-4.	Partition Status Display . . . . .	9-12
9-5.	System Initialization File for a Single-Processor Workstation . . . . .	9-17
9-6.	System Initialization File for a Multiprocessor Workstation . . . . .	9-21
9-7.	System Initialization File for an SRP . . . . .	9-23
10-1.	CFA Configure Display . . . . .	10-2

## Contents

---

11-1.	Workstation Disk Type and Bad Spot Report .....	11-3
11-2.	Disks in an SRP Primary Cabinet (Rear View) .....	11-5
11-3.	Bad Spots File for a Non-SCSI Disk .....	11-11
11-4.	Format Template .....	11-14
11-5.	Device Template .....	11-24
11-6.	Volume Status Display .....	11-34
12-1.	Write-Enable Ring on a Half-Inch Tape .....	12-6
14-1.	File-System Cache .....	14-2
16-1.	Operating System Configuration File for a Single-Processor Workstation .....	16-3
16-2.	Operating System Configuration File for a Multiprocessor Workstation .....	16-4
16-3.	Operating System Configuration File for an SRP .....	16-5
18-1.	SignOn Screen: Conventional Interface .....	18-6
18-2.	SignOn Screen: Window Interface .....	18-6
19-1.	SRP Slot Numbers (in Hexadecimal) .....	19-4
19-2.	SRP Real-Mode Processor Status LEDs .....	19-17
19-3.	LED Patterns for Hexadecimal Digits .....	19-18
19-4.	LED Sequence Pattern for a Hexadecimal Error Code .....	19-19
19-5.	SRP Protected-Mode Processor Status LED .....	19-20
19-6.	Sequence Pattern for a Decimal Error Code .....	19-20
19-7.	Hexadecimal-to-Decimal Conversion Chart .....	19-40

# Tables

2-1.	SRP Cabinet Summary .....	2-7
3-1.	Workstation Operating Systems .....	3-3
3-2.	SRP Operating Systems .....	3-5
4-1.	Editor Application Keystrokes .....	4-7
4-2.	Editor Files Menu Options .....	4-9
4-3.	Cluster View Parameters .....	4-14
5-1.	Workstation-Type Numbers .....	5-4
6-1.	Change Volume Name Parameters .....	6-5
6-2.	Protection and Access Levels .....	6-6
6-3.	Create Directory Parameters .....	6-7
6-4.	Set Directory Protection Parameters .....	6-8
6-5.	Set Protection Parameters .....	6-10
7-1.	User File Keywords Recognized by SignOn .....	7-9
8-1.	Installation Manager Command-Form Parameters .....	8-7
8-2.	Installation Manager Parameters-Form Selections .....	8-8
8-3.	Installation Manager Interactive Fields .....	8-9
8-4.	Software Installation Errors .....	8-15
9-1.	System Service Commands and Run Files .....	9-15
9-2.	JCL Statements for System Initialization Files .....	9-18
9-3.	System Service Errors .....	9-30
10-1.	Install CFA Workstation Agent Parameters .....	10-4
11-1.	Device Template and Bad Spot Requirements for Disks .....	11-6
11-2.	Format Disk Parameter Fields .....	11-7
11-3.	Format Disk Standard Format Templates .....	11-15
11-4.	Format Disk Standard Device Templates .....	11-24

## Contents

---

12-1.	Approved Tapes for Data Storage . . . . .	12-2
12-2.	Sequential Access Service Parameters . . . . .	12-4
12-3.	CD-ROM Service Parameters . . . . .	12-8
13-1.	Volume Archive Parameters . . . . .	13-4
13-2.	Restore Archive Parameters . . . . .	13-10
17-1.	CTOS I Prefix Files . . . . .	17-5
17-2.	CTOS II Prefix Files . . . . .	17-6
17-3.	CTOS III Prefix Files . . . . .	17-6
17-4.	CTOS/XE Prefix Files . . . . .	17-7
19-1.	Create Backup Floppy Parameters . . . . .	19-35
A-1.	Boot ROM Error Code E0h . . . . .	A-5
A-2.	Boot ROM Error Code E1h . . . . .	A-6
B-1.	SRP General Processor Display Activity . . . . .	B-2
B-2.	SRP 186 Processor Activity . . . . .	B-3
B-3.	System Crash Status Words . . . . .	B-14
B-4.	General Status Register Contents . . . . .	B-15
C-1.	Color-Bit Combinations . . . . .	C-5
C-2.	Color Mapping: CTOS to Standard VGA . . . . .	C-6

# About This Guide

## Introduction

The *CTOS System Administration Guide* contains information about the CTOS® operating system and its associated utilities, called Standard Software. This edition of the *CTOS System Administration Guide* is relative to CTOS I 3.4, CTOS II 3.4.4, CTOS III 1.1, CTOS/XE 3.4.100, Standard Software 12.3, Video Access Method 4.2, and the System Build Utilities 1.1/3.4.

## What This Guide Covers

This guide describes the tasks performed by system administrators and provides detailed procedures for performing those tasks in a CTOS-based cluster. It also provides an overview of workstation and shared resource processor (SRP™) hardware and software components and introduces the software tools available to system administrators. In addition, it describes troubleshooting techniques for both workstations and SRPs.

This guide does not address specific hardware issues, such as the installation of workstations, SRPs, or cluster cabling. See the appropriate installation guides for information about your hardware products. Installation of cluster hardware is described in the *CTOS Cluster and Network Hardware Installation Guide*.

See the *CTOS System Software Installation Guide* for information about installing the operating system and Standard Software. See the Software Release Announcements for hardware requirements and information about software compatibility.

# Who Should Use This Guide

This manual is for system administrators who are responsible for setting up and maintaining CTOS-based clusters. You may be an experienced full-time system administrator, or you may administer a cluster in addition to other job duties. In either case, this manual is written with the following assumptions about your skills and knowledge:

- You are familiar with basic workstation operations, such as signing on and using the Executive.
- You understand file system concepts, such as volumes, directories, and *!Sys*.

If you need to learn about basic workstation operations or the file system, see the *CTOS Executive User's Guide* before you continue with this manual.

# How This Guide Is Organized

The *CTOS System Administration Guide* is organized as follows:

## **Section 1. If You Are New to System Administration**

This section provides an overview of administrative tasks for those who are new to system administration.

## **Section 2. Understanding Hardware**

This section presents an overview of workstation and SRP hardware.

## **Section 3. Understanding System Software**

This section presents an overview of the operating systems and Standard Software utilities.

## **Section 4. Using Administrative Tools**

This section describes the software commands and applications you use for system administration.

## **Section 5. Bootstrapping**

This section describes the bootstrap and system initialization sequences of workstations and SRPs.

## **Section 6. Implementing System Security**

This section describes how to password-protect your system and prevent unauthorized persons from using it.

## **Section 7. Customizing User Environments**

This section describes how to define the commands and applications available to users.

## **Section 8. Installing Applications**

This section describes some of the most commonly used software applications and provides general information about installing them.

## **Section 9. Installing System Services**

This section describes how to install loadable system services, which supplement the operating system by providing access to additional resources, such as tape drives, modems, and printers.

## **Section 10. Accessing Data Throughout the Cluster**

This section describes how to install and use the Cluster File Access facility, which allows files to be shared among client workstations.

## **Section 11. Adding Hard Disks**

This section describes how to format disks for use on workstations and SRPs.

## **Section 12. Using Tape and CD-ROM Drives**

This section describes the media and system services required to use quarter-inch cartridge (QIC), digital data storage (DDS), half-inch tape, and CD-ROM drives.

## **Section 13. Backing Up and Restoring Data**

This section describes how to back up and restore disks to and from tape archive media.

## **Section 14. Using a File System Cache**

This section describes how to configure a file system cache, which increases file access speed.



### **Section 15. Optimizing System Performance**

This section describes how to improve system performance by making optimal use of memory, disk space, and other system resources.

### **Section 16. Configuring Operating Systems**

This section describes how to configure workstation and SRP operating systems to function optimally in your environment.

### **Section 17. Building a Customized Operating System**

This section describes how to further configure an operating system with the Mini-Customizer utility or by building a customized version.

### **Section 18. Customizing Standard Software**

This section describes how to customize screen messages and color attributes for SignOn and the Standard Software utilities.

### **Section 19. Troubleshooting**

This section describes troubleshooting tips and techniques.

### **Appendix A. Workstation Bootstrap and Initialization Status Codes**

This appendix lists status codes that are displayed by workstations during bootstrapping and system initialization.

### **Appendix B. Shared Resource Processor Front Panel and Processor Codes**

This appendix lists status codes that are displayed by SRPs on the front panel and processor LEDs.

### **Appendix C. Configuring Standard VGA**

This appendix describes how to configure character attributes, such as bold, underline, and color, on EISA/ISA workstations that are not equipped with a CTOS VideoCard.

### **Appendix D. CTOS I/O Processor Card Error Codes**

This appendix lists diagnostic error codes that are written to the system log file if CTOS I/O Processor cards fail.

## What Is New in This Guide

Significant changes in the content and organization of this guide are described below.

- A new subsection, “Understanding EISA/ISA Workstations,” has been added to Section 2, “Understanding Hardware.”
- Documentation for the Editor application has been added to Section 4, “Using Administrative Tools.”
- Videoless workstation servers are documented in Section 6, “Implementing System Security.”
- Information about formatting a partitioned disk and activating disk partitions has been added to Section 11, “Adding Hard Disks.”
- Section 12 has been changed from “Using Tape Drives” to “Using Tape and CD-ROM Drives” and includes information about installing and configuring CD-ROM software.
- “Configuring Workstation Operating Systems” and “Configuring Shared Resource Processor Operating Systems,” which were separate sections in the previous edition of this manual, have been combined into Section 16, “Configuring Operating Systems.”
- Appendixes A through C have been added. See “How This Guide Is Organized,” above, for a description of each appendix.

In addition, documentation for CTOS III, which now supports multi-processor workstation servers, is included throughout this manual.

## Terminology Used in This Guide

The following terms have been changed or added since the previous edition of this guide:

Client workstation	Replaces the term <i>cluster workstation</i> . The term <i>cluster</i> , however, is still used to describe the CTOS local area network.
CTOS I/O Processor Card (IOP)	An expansion card for EISA workstations that contains its own processor.

Multi-processor workstation

An EISA workstation that contains one or more CTOS I/O Processor Cards.

Notebook workstation

A small portable workstation that can be easily detached and then reconnected to the cluster.

Other new terms are defined in the sections to which they apply. See also Section 1, "If you Are New to System Administration," for a list of more terms that pertain to CTOS system administration.

## Conventions Used in This Guide

The following conventions are used throughout this manual:

- New terms appear in italics and are defined in the glossary.
- Command names are capitalized and appear in **bold** type, for example, **Format Disk** and **Volume Status**.
- Names of fields appear in *italics*, for example, *[Password]* field.
- Variable information also appears in *italics*, for example, *...WsNNN*, where *NNN* stands for a three-digit number.
- Names of keys appear in **UPPERCASE BOLD** type, for example, **GO**.
- Sometimes you need to issue keystroke combinations by holding down one key while you press another; such a sequence is shown as follows:  
**CODE+DELETE**
- Volume, directory, and file names appear in *italics*, for example, *[Sys]<Sys>Config.sys*.

## Where to Find More Information

The following manuals are included in the documentation set for Standard Software:

### ***CTOS Executive Reference Manual***

This manual documents the Executive command interpreter, which is a primary tool for system administration. It describes Executive commands packaged with Standard Software and is arranged alphabetically by command name. Keep it handy as a companion volume to the this manual.

### ***CTOS Executive User's Guide***

This user's guide provides step-by-step procedures for the most commonly used commands and features of the Executive. It is a good resource for people who are new to CTOS or are occasional users of the Executive.

### ***CTOS System Software Installation Guide***

This manual assists with the software installation process of CTOS I, CTOS II, CTOS III, CTOS/XE, Standard Software, and the Video Access Method (VAM).

### ***CTOS Batch Manager II Installation, Configuration, and Programming Guide***

This manual describes the Batch Foreground command that is packaged with the Standard Software utilities. In addition, it describes the Background Batch function of Batch Manager II, which is packaged and installed separately from the Standard Software utilities.

## Hardware Installation Guides

See the following manuals for information about workstation hardware configurations:

### ***CTOS Media User's Guide***

### ***CTOS Workstations Planning and Owner's Maintenance Guide***

### ***CTOS Cluster and Network Hardware Installation Guide***

***CTOS XE-530 Shared Resource Processor Hardware Installation Guide***

***OpenSystems Workstations Hardware Installation and Owner's Guide***

***SuperGen Series 3000 Hardware Installation and Owner's Guide***

***OpenSystems Workstations SuperGen Series 3400 Hardware Installation and Owner's Guide***

***SuperGen Series 5000 Installation and Maintenance Guide***

## **Administration Guides for Applications**

See the following manuals for information about Context Manager, printing, networking, and electronic mail products:

***CTOS Generic Print System™ Administration Guide***

***CTOS Context/Window Manager Installation and Configuration Guide***

***CTOS BNet II Installation, Configuration, and Administration Guide***

***CTOS BNet II Cluster Access Administration Guide***

***CTOS LANce/pc and LANce/LM Administration and Programming Guide***

***CTOS OFIS™ Mail Administration Guide***

## **Software Development Manuals**

See the following manuals for information about CTOS software development products:

***CTOS Debugger User's Guide***

***CTOS Editor User's Guide***

***CTOS Operating System Concepts Manual***

***CTOS Procedural Interface Reference Manual***

***CTOS Programming Guide***

***CTOS Programming Utilities Reference Manuals***

# Section 1

## If You Are New to System Administration

### Overview

If you are new to system administration or have never worked with CTOS, you may not be familiar with certain terminology or the various tasks that a system administrator performs.

This section defines some terms and briefly describes a number of administrative tasks. After reading this section, you should have a good idea of what is required of you as a system administrator.

The sections that follow expand on the terms and concepts presented here and provide detailed procedures for performing specific tasks.

### Terms

The terms defined below are used throughout this manual and in most of the other documentation you will use with CTOS-based systems. These terms, and many others, are described in more detail in later sections.

Application	A program you interact with on your workstation. This term usually refers to a multifunctional program, such as a word processing or accounting package.
Client workstation	A workstation that is connected to and communicates with a server (formerly called <i>cluster workstation</i> ).
Cluster	A group of computers connected together so that they can share resources, such as files, printers, or databases.

Configuration	An arrangement of parts, such as computer hardware, or of elements, such as software programs, or a combination of both. (See also “What Does ‘Configuration’ Mean?,” below).
CTOS	The overall name for the family of workstation and shared resource processor operating systems.
CTOS I	The real-mode workstation operating systems.
CTOS II	The protected-mode workstation operating systems without virtual memory.
CTOS III	The protected-mode workstation operating systems with virtual memory.
CTOS I/O Processor Card (IOP)	An installable card for EISA workstations that contains its own processor.
CTOS/XE	The shared resource processor operating systems.
Multiprocessor workstation	An EISA workstation that contains a CTOS I/O Processor Card.
Notebook workstation	A small portable workstation that can be easily detached and then reconnected to the cluster.
Operating system	The program that controls data input and output and execution of application programs on a computer.
Server	One computer within a cluster that controls resources that are shared throughout the cluster.
Shared resource processor (SRP)	A floor-model computer that functions as a server (also called an XE).
Standard Software	A set of commands and applications you use to perform basic tasks, such as copying files or backing up a disk.



System services	Optional programs that expand the capabilities of the operating system.
User file	A configuration file that defines the working environment for a specified user name.
Workstation	A desktop computer that may be configured as a server, a client, or a standalone workstation.

## What Does “Configuration” Mean?

The word *configuration* is a commonly used term in system administration. It is used in a variety of circumstances, as described below:

Hardware configuration	The hardware devices that are present on a workstation or SRP.
Cluster configuration	The combination of different systems in a cluster.
Software configuration	The combination of the operating system and applications installed on a system.
Configuration file	A file that contains parameters for a software product or a hardware device.

CTOS systems are very configurable. That means that each individual workstation or SRP can be set up in many different ways. Configurability allows systems within the cluster to meet the needs of individual users.

## Administrative Duties

The actual duties you perform as a system administrator depend a great deal on the size of your cluster. In general, you are responsible for the server; sometimes you are also responsible for other workstations in the cluster. For example, if you are administering a small cluster in addition to your other job duties, it is likely that other users in the cluster administer their own workstations. On the other hand, if you are administering one or more large clusters as a full-time job, most likely you will be responsible for all aspects of system administration.

The tasks described below are typical administrative duties. In addition, many system administrators are also responsible for hardware maintenance. See the appropriate installation guides and technical reference manuals for help with your hardware requirements.

### Setting Up a System

The following list briefly describes the tasks associated with setting up a workstation or SRP. See the cross-referenced sections and manuals for more detailed information.

1. Assemble the hardware.

Setting up workstation hardware is a simple task; the only tool you need is a small, flathead screwdriver for attaching the monitor cable to the workstation. See the installation guide that accompanies the workstation for detailed instructions. SRPs are usually installed by a customer service engineer; for detailed information, however, see the *XE-530 Shared Resource Processor Hardware Installation Guide*.

2. Install system software.

Before a workstation or SRP will function as a computer, you must install system software, which consists of the operating system and Standard Software. See the *CTOS System Software Installation Guide* and the Software Release Announcements for the operating systems and Standard Software.

3. Configure the operating system.

You may need to configure the operating system to recognize all hardware components on the system. See Section 16, "Configuring Operating Systems."

4. Connect workstations to the server.

To form a cluster, workstations are connected to the server. See the *CTOS Cluster and Network Hardware Installation Guide* for detailed information about connecting the systems in a cluster.

### 5. Install applications.

After you install system software, you install applications. See Section 8, “Installing Applications,” for general information. See the release documentation or the software installation guide for each application for specific instructions.

### 6. Install system services.

In many cases, system services are required to run applications. See Section 9, “Installing System Services,” and the release documentation for each application.

### 7. Configure applications.

Some applications have configuration files that affect the way they work. See the release documentation and the manual for the application for more information.

## Other Administrative Tasks

In addition to setting up workstations, system administrators frequently are responsible for the following:

Assigning  
passwords

As system administrator, you are responsible for password-protecting the server.

Creating  
directories

In some cases, particularly if users share disk space on the server, you are responsible for creating directories.

Creating user files

You create user files to customize the working environment for each user.

Adding disks

In many cases, you are responsible for planning the use of disk space and adding disks when necessary.

Updating software

You install new software applications and update existing ones when new versions are released.

## If You Are New to System Administration

---

Configuring hardware and software

Whenever you add hardware or software, check the release documentation and application manuals for specific configuration requirements.

Performing backups

You are usually responsible for backing up disks on the server, and in some cases, all disks in the cluster. You do this periodically (in many cases, daily) to make duplicate copies of all data stored on the system.

Monitoring the system

As you become more experienced, you might want to keep track of system activity, for example, the number of users who are signed on at certain times or how quickly disks fill up. This helps you tailor the system to the needs of your workplace.

Correcting problems

You are responsible for diagnosing and solving problems. Troubleshooting is one of the most challenging aspects of system administration.

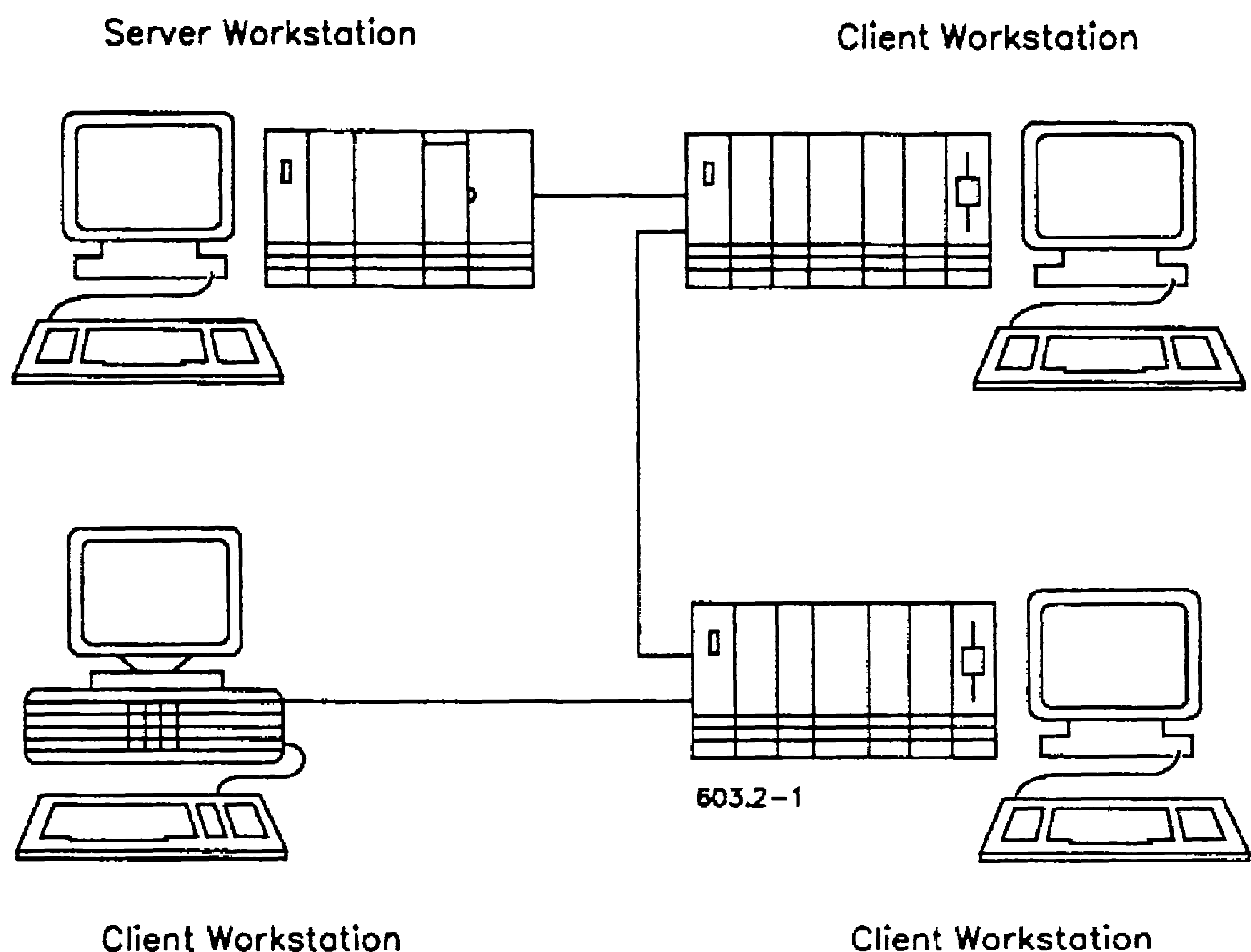
## Section 2

# Understanding Hardware

## What Is a Cluster?

A *cluster* is a group of computers that are connected together so that they can share disks, printers, and other system resources. A single computer within the cluster functions as a *server*, which controls resources for the entire cluster. The server and clustered workstations (called *client workstations*) are connected either with cables or with TeleCluster™, which connects them via telephone lines. Figure 2-1 shows a simple cluster configuration.

Figure 2-1. Simple Cluster Hardware Configuration



The server can be either a workstation, as shown in Figure 2-1, or a larger computer called a shared resource processor (SRP). This section provides an overview of both workstation and SRP hardware components, which are referred to throughout this manual. See the appropriate hardware installation guides for more specific information. For details about setting up and connecting systems to a cluster, see the *CTOS Cluster and Network Hardware Installation Guide*.

## Workstation Hardware

A workstation is a desktop unit that functions as a server, a client workstation, or a standalone computer. A workstation consists of a central processing unit (CPU), a keyboard, and a monitor. Optional modules or expansion cards can be attached to most workstations to provide local disk storage space, graphics, and other enhancements.

### Workstation Models

Although there are many distinct workstation models, they can be grouped into the following categories. For more specific information, see the installation guide for your particular workstation.

#### X-Bus Workstations

An X-Bus™ workstation is a collection of separately housed modules, each containing an individual hardware component, such as a processor, a disk, or a graphics controller. X-Bus workstations include the B26, B28, and B38 models.

A variation is the B39 model X-Bus workstation, which contains the processor, a hard disk, and a floppy disk drive in the base unit, and can be expanded with X-Bus modules.



### X-Bus+ Workstations

An X-Bus+ workstation consists of a base unit containing the CPU and removable cartridges, which house disk drives, graphics controllers, tape drives, and other optional equipment. In addition, an X-Bus+ workstation can be equipped with an adapter to accept X-Bus-style SCSI disk and tape modules. The SuperGen Series 5000 is an X-Bus+ workstation.

### EISA and ISA Workstations

EISA and ISA workstations consist of a base unit containing a CPU and other hardware components, such as a hard disk, a floppy disk drive, and a video controller. They can be upgraded with “industry standard” hardware components. The SuperGen Series 3000 and some Series 2000 models are ISA workstations; the SuperGen Series 4000, 6000, and 7000 are EISA workstations. See “Understanding EISA/ISA Workstations,” later in this section, for more detailed information.

### Diskless Workstations

A diskless workstation consists of a base unit containing the CPU and video controller. It uses disks on the server for data storage; disks are not connected directly to it. Some diskless workstations can be upgraded with expansion cards for graphics or extra memory.

### Processors

Workstations are equipped with Intel® central processing units (CPUs). For most system administration purposes, 80286 and higher CPUs are called *protected-mode* processors, and 80186 CPUs are called *real-mode* processors. Those terms, however, actually describe the operating system that is running on the workstation. See Section 3, “Understanding System Software,” for information about real-mode and protected-mode operating systems.

### CTOS I/O Processor Cards

Most EISA workstations can be equipped with CTOS I/O Processor Cards (IOPs), which add multiprocessing functionality to the workstation. An IOP contains its own CPU and can significantly improve performance of the workstation. For example, an IOP may be used to control network communications, thereby freeing processor time for disk I/O activities. IOPs are described below.

Cluster IOP	The Cluster IOP provides one RS-485 cluster channel (two RJ-45 connectors), which supports 16 workstations.
Communications IOP	The Communications IOP provides two RS-232-C ports with one 25-pin and one 15-pin connector.
Ethernet IOP	The Ethernet IOP provides one network communications channel and is equipped with one AUI 15-pin connector, one thinnet BNC connector, and one 10BaseT RJ-45 connector.
ISDN IOP	The ISDN IOP provides one 2B+D basic rate ISDN channel with one RJ-45 connector.

### Processor Identifiers

When IOPs are installed in a workstation, the operating system assigns a unique identifier (ID) to each processor. The master processor is identified as EP00 (EP stands for EISA-bus processor); IOPs are then identified by the two-letter acronym EP and a two-digit number identifying the IOP's position along the bus, for example, *EP01* and *EP02*.

You use processor IDs when you configure software. You will also encounter them when using the **System Manager** command or reading the system error log.



## Cartridges, Modules, and Expansion Cards

Many cartridges, modules, and other expansion components are available for workstations. The following list describes only a few. See your installation guide for information about installing modules and expansion cards.

Disks	Various capacity hard disks, as well as 5-1/4 and 3-1/2 inch floppy drives, are available for most workstation models.
Video controllers	A video controller other than the one that is included with the workstation processor may be required to use certain high-resolution monitors.
Tape drives	Quarter-inch cartridge (QIC) and digital data storage (DDS) tape drives are available for X-Bus, X-Bus+, and EISA workstations.
CD-ROM drives	Compact disc (CD) drives are available for X-Bus, X-Bus+, and EISA workstations.

## Monitors

Major categories of monitors are listed below:

- Green-on-black monochrome monitors
- Color graphics monitors
- High-resolution bit-map monitors
- Video graphics array (VGA) monitors

Not all monitors can be used with every workstation model, and in many cases, an additional video controller is required. See the workstation installation guide to find out about monitor and graphics-controller compatibility.

### Keyboards

Several keyboard models are available for workstations. Although the position of some keys may vary, most are labeled with CTOS key names, which are used consistently throughout this manual.

## Shared Resource Processor Hardware

The shared resource processor (SRP) is a floor-model computer containing multiple processors that perform different functions. For example, some processors control disk operations, while others control network communications. The processors constantly pass data back and forth and are dependent on one another to function as a system. Many processor combinations are available to provide optimal hardware configurations.

An SRP always functions as a server. It does not have a keyboard and monitor; therefore, a special utility, called Cluster View, is required to communicate with SRP processors. (Cluster View is described in Section 4, "Using Administrative Tools.")

Hardware components of the SRP are briefly described below. For more detailed information, see the installation guide for your particular model.

### Cabinets

SRP hardware components are housed in cabinets (sometimes called enclosures). An SRP consists of one primary cabinet and up to five secondary cabinets. A primary cabinet, as pictured in Figure 2-2, is equipped with a keyswitch, a reset button, a two-numeral status display, and a QIC tape drive. Both primary and secondary cabinets contain processor boards and disk drives. The various cabinet models are summarized in Table 2-1.

Figure 2-2. SRP Primary Cabinet

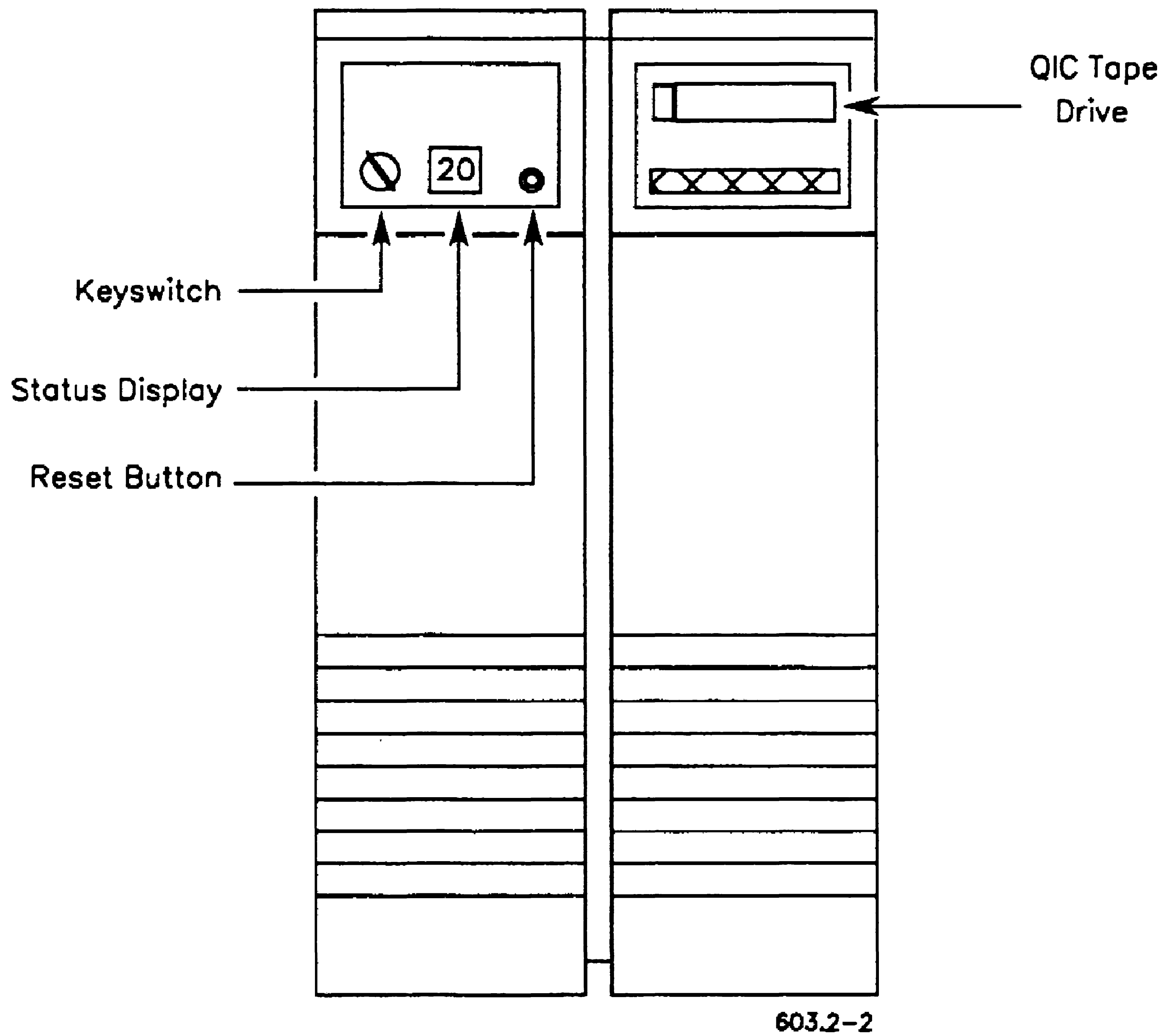


Table 2-1. SRP Cabinet Summary

Name	Type	Description
C-Box	Primary cabinet	Contains a QIC tape drive, three 5-1/4 inch hard disk drive slots, and six processor board slots.
E-Box	Primary cabinet	Contains two 8-inch hard disk drive slots, six processor board slots, and optionally, a QIC tape drive.
B-Box	Expansion cabinet	Contains four 5-1/4 inch hard disk drive slots and six processor board slots.
X-Box	Expansion cabinet	Accommodates ten 5-1/4 or three 8-inch hard disk drives.

### Processor Boards

The latest model SRP processor boards are equipped with Intel 80386 CPUs. Older processor boards contain Intel 80186 CPUs. In most cases, the terms *real mode* and *protected mode* are used throughout this manual to differentiate between the 80186 and 80386 SRP processors, respectively. See also Section 3, "Understanding System Software," for information about real-mode and protected-mode operating systems.

To identify SRP processor boards, look at the acronyms stamped on the back, which are visible when you open the rear cabinet doors. Those acronyms are used throughout this manual. Your SRP may not contain every type of processor.

### Protected-Mode Processors

Late model and new SRPs are equipped with protected-mode processors. A protected-mode processor may consist of either a processor board alone or a processor board in combination with a special-purpose interface board. Protected-mode processor are described below.

General Processor  
(GP)

The GP is equipped with two RS-485 cluster channels (four connectors), which can support up to 32 workstations (16 per channel), two RS-232-C ports, and one parallel printer port.

General Processor  
with SCSI Interface  
(GP+SI)

The GP+SI consists of a General Processor and a SCSI Interface board (SCSI stands for Small Computer System Interface). The interface board is equipped with two SCSI device controllers, which can support up to eight SCSI devices.

General Processor  
with Comm-  
unications Interface  
(GP+CI)

The GP+CI consists of a General Processor and a Communications Interface board. The interface board is equipped with six RS-232-C ports.

General Processor  
with Ethernet  
Interface (GP+EI)

The GP+EI consists of a General Processor and an Ethernet Interface board. The interface board is equipped with AUI (attachment unit interface), thinnet (thin ethernet), and 10BaseT connectors.

## Real-Mode Processors

Although newly installed SRPs are equipped with protected-mode processors only, upgraded systems may contain a combination of real-mode and protected-mode processors. Real-mode processor boards are described below. See the *XE-530 Shared Resource Processor Hardware Installation Guide* for information about the compatibility of different processor models.

Cluster Processor (CP)	The CP contains two RS-422 cluster channels (four connectors), which can support up to 16 workstations (8 per channel); two RS-232-C ports; and one parallel printer port.
File Processor (FP)	The FP contains four disk interface ports, which can support up to four ST-506 hard disk drives.
Data Processor (DP)	The DP consists of a Storage Processor (see below) with a Storage Controller board (SC) in the adjacent slot. A DP controls up to six external SMD disk drives.
Storage Processor (SP)	The SP contains a tape interface that supports up to four external 9-track half-inch tape drives.
Terminal Processor (TP)	The TP contains ten RS-232-C ports and one parallel printer port.

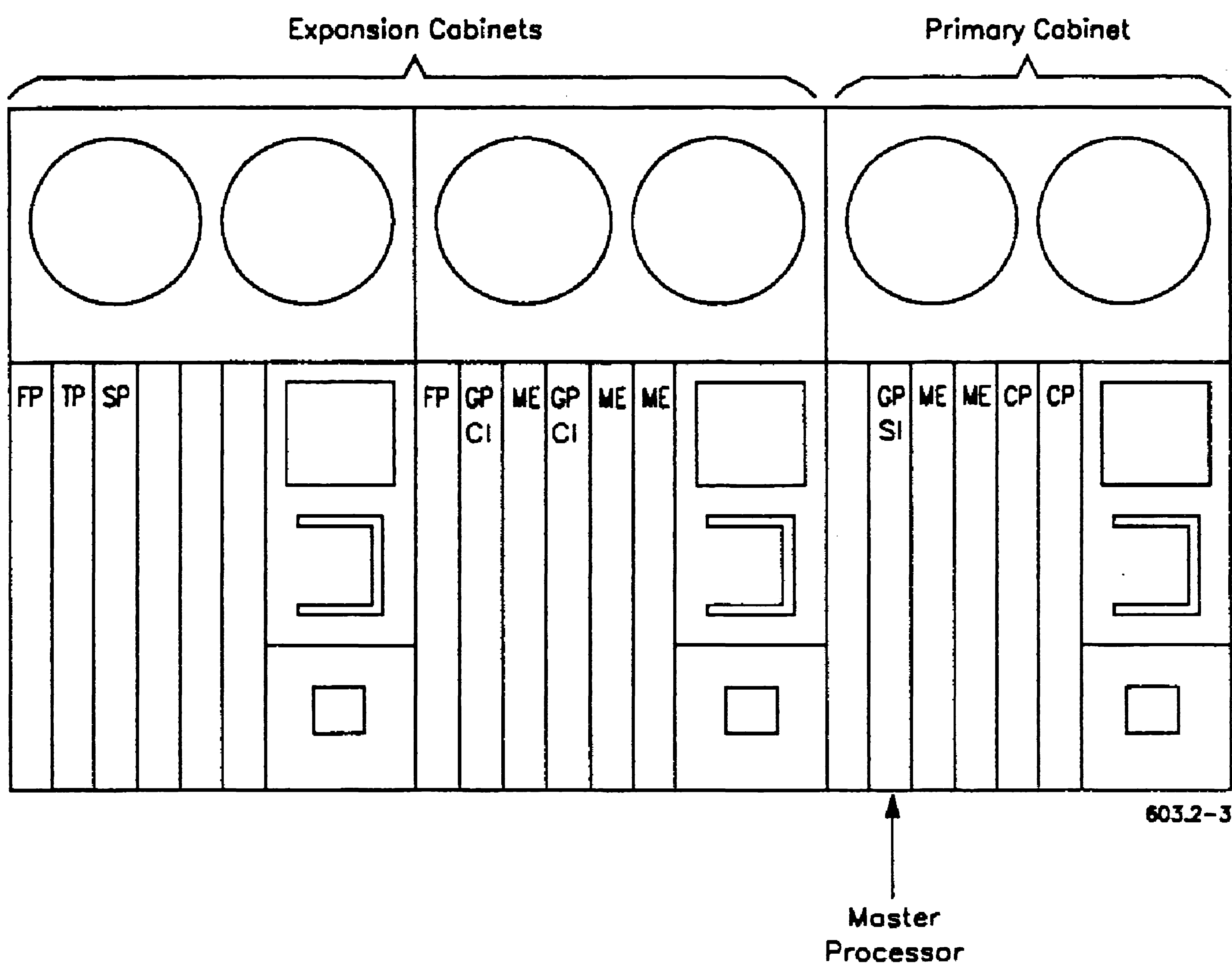
## The Master Processor

SRPs contain a master processor, which bootstraps itself first and then controls booting of the other processors. The master processor is the first processor in the primary cabinet. It must be a disk-controlling processor, that is, a General Processor with SCSI Interface, a File Processor, or a Data Processor, as described earlier in this section.

Figure 2-3 shows the location of the master processor within a three-cabinet SRP.



Figure 2-3. SRP Master Processor



## Processor Identifiers

A unique identifier (ID) is assigned by the operating system to each processor. It consists of the two-letter acronym for the processor and a two-digit number identifying the processor's position within the SRP, for example, *GP00* or *CP01*.

You use processor IDs when you are configuring software; you will also encounter them when using the System Manager or reading the system error log. Processor numbering schemes are described below.

## Protected-Mode Processor Numbers

All protected-mode processors are identified by the acronym GP and are numbered sequentially, regardless of the presence of a special-purpose interface. For example, in Figure 2-4, the first GP, which has a SCSI Interface, is GP00; the second GP, which has a Communications Interface, is GP01, and so on.

## Real-Mode Processor Numbers

Each type of real-mode processor has a unique acronym, for example, FP for File Processor or CP for Cluster Processor. Processors of the same type are numbered independently of other processor types. When viewing an SRP from the back, as shown in Figure 2-4, processor numbering begins from the left of the primary cabinet. For example, the first File Processor is identified as FP00, while the second is FP01; the first Cluster Processor is CP00, while the second and third are CP01, CP02, and so on. Note in Figure 2-4 that sequentially numbered processors are not necessarily adjacent to one another (for example, FP00 and FP01).

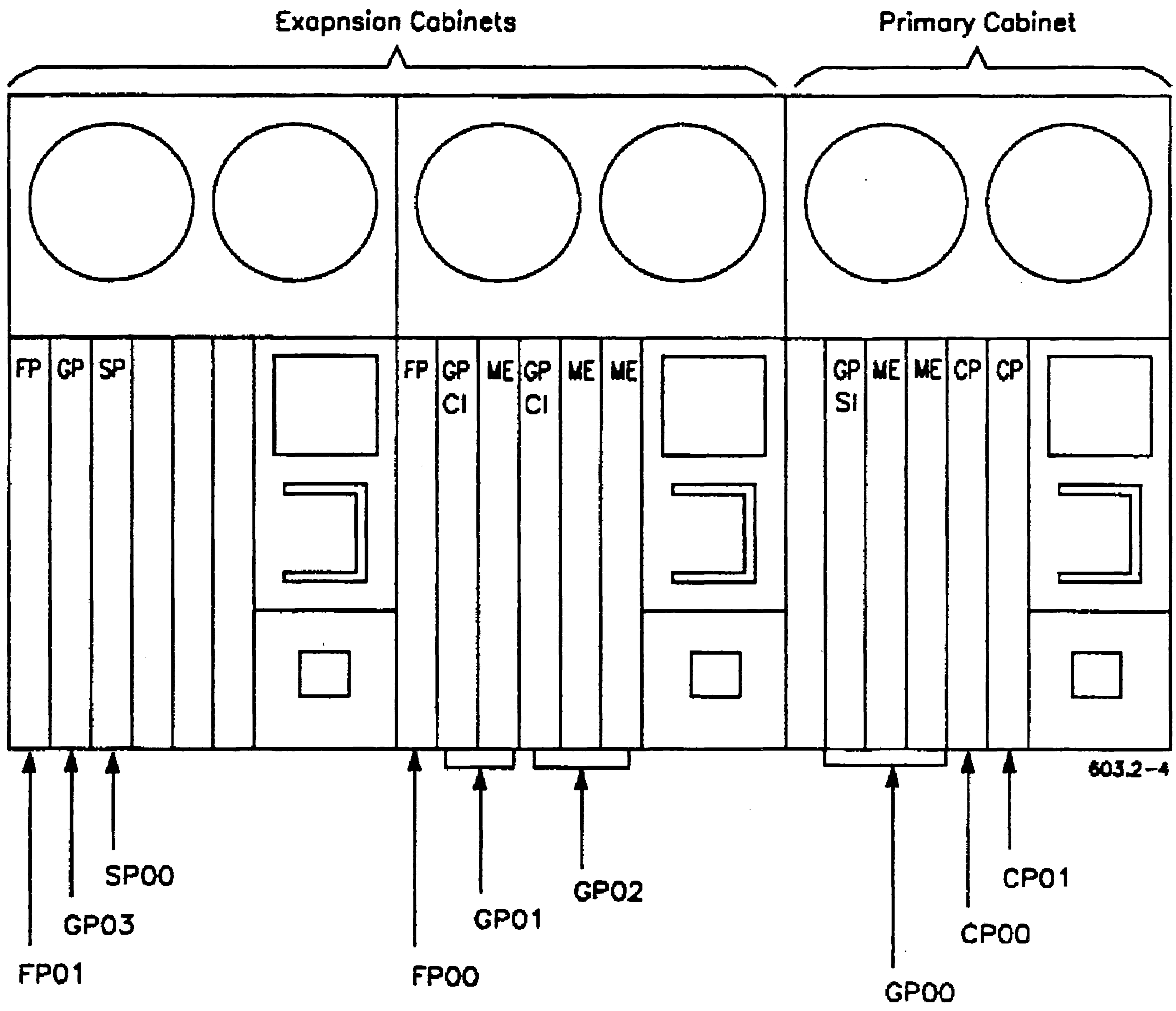
## Memory Expansion Boards

Memory expansion boards (ME), as shown in Figure 2-4, are available for protected-mode processors and some real-mode processors. They expand the amount of RAM available on the processor.

Protected-mode processors can be expanded with two memory expansion boards for a total of 64M bytes of memory. See the *XE-530 Shared Resource Processor Hardware Installation Guide* for details.

Real-mode processors containing 256K bytes of memory can be expanded with one memory expansion board for a total of 768K bytes of memory. Memory expansions cannot be added to real-mode processors containing 786K bytes of memory.

Figure 2-4. SRP Processor Identifiers





## Understanding EISA/ISA Workstations

This section is written for CTOS system administrators who are new to industry-standard EISA/ISA systems. Recently, the CTOS operating system has been modified to run on EISA/ISA hardware, where before, it was designed to run exclusively on proprietary hardware.

While CTOS functions the same as before on industry-standard hardware, some of the features of the hardware itself are different from X-Bus style CTOS workstations.

### Industry Standard Hardware

EISA/ISA workstations consist of industry-standard hardware components. *ISA* stands for industry-standard architecture; *EISA* stands for extended industry-standard architecture. EISA and ISA hardware components are designed and manufactured by many different companies, yet because they are designed according to EISA and ISA standards, they can be purchased and used interchangeably.

### EISA and ISA Cards

A feature of EISA/ISA workstations is their expandability by means of adding cards. The ISA bus and EISA bus contain slots that accept expansion cards. ISA and EISA cards are distinguishable from one another by the connector edge that fits into the slot on the workstation. See the *OpenSystems Workstations Hardware Installation and Owner's Guide* for a picture.

An EISA workstation accepts both EISA and ISA cards, while an ISA workstation accepts ISA cards only. The Client Card and the CTOS VideoCard are examples of ISA cards. CTOS IOPs are examples of EISA cards. Other EISA and ISA cards, such as disk controllers or communications interfaces, may also be available for your system.

Both EISA and ISA cards require special configuration. With EISA cards a special product identifier makes the configuration somewhat simpler than with ISA cards. Configuration is discussed in more detail later in this section.

### EISA and ISA Workstation Models

Unisys sells and supports the following EISA and ISA workstations:

Some Series 2000  
and Series 3000

Entry-level ISA systems designed for use as client workstations.

Series 4000

Mid-range EISA systems designed for use as either server or client workstations.

Series 6000 and  
7000

High-end EISA servers.

See the OpenSystems Workstations hardware installation guides for more detailed information.

### What Is the BIOS?

The term *BIOS* stands for basic input/output system. For CTOS system administration purposes, the BIOS is roughly equivalent to the boot ROM. It executes hardware tests, locates and boots the operating system, and then transfers control to the operating system.

In addition to its boot ROM functions, however, the BIOS is a firmware program that provides device-level programmatic interfaces. The predictable functionality of the BIOS at the device level is an important component of industry-standard hardware.

As a system administrator, your interaction with the BIOS is not required. In rare instances, you might need to “flash” the BIOS, which means to overwrite a portion of it with new information. In such a case, however, you would be carrying out instructions from a Technical Support engineer or from detailed documentation.

### What Is CMOS?

The term *CMOS* stands for complementary metal-oxide semiconductor and refers to battery backed-up RAM. The contents of battery backed-up RAM remain intact when the system is turned off or in case of a power failure.

### Hardware Configuration Programs

A major difference between X-Bus style and EISA/ISA workstations is that you must configure hardware components on EISA/ISA workstations to ensure that the components do not conflict with each other. This is required when you set up a new system or when you add or remove components.

To configure the hardware on ISA workstations you use a configuration program that is called Setup. On EISA workstations you use a similar program called the System Configuration Utility (SCU). The Setup program and SCU record hardware configuration information in CMOS.

#### Setup

The Setup program is stored in the ROM/BIOS on ISA systems and notebook workstations, so it is always available for checking system-configuration data. See the hardware installation guide for your ISA workstation for information about using Setup.

#### System Configuration Utility

On EISA systems the SCU is provided on floppy disks. It is described in the SCU manual that comes with your system.

When an EISA system boots if there is any discrepancy between the actual hardware that is detected and the hardware configuration information in CMOS, you see a message that indicates the problem. You may be prompted explicitly to run the configuration utility, or you may get the following message:

```
Hit F1 to continue.
```

At that point you insert the SCU disk and go through the configuration process. The SCU is usually used with configuration files that come with the cards and other special function hardware you add to your system. It prompts you to mount floppy disks that contain those files when needed.

When you have finished configuring your new EISA system, be sure to save the SCU disks and any additional configuration file disks that you may have in a safe place near the machine so that you can find them again if additional configuration is required.

### Disks

When you add a disk to an EISA/ISA workstation, you might need to configure the hardware or the software. General information about configuring SCSI and IDE drives is included below. See the hardware installation guide for your workstation for information about installing the drive.

### SCSI Drives

Before you add a SCSI drive, set its jumpers to a unique SCSI ID. Information displayed during booting informs you about the SCSI IDs that are already used. See the documentation for the drive for information about jumper settings.

You do not need to configure SCSI drives with Setup or the SCU.

### IDE Drives

Before you install an IDE drive, you may need to set its jumpers; see the documentation that accompanies the drive. After you install the IDE drive, you need to configure it with the Setup program or the SCU. See the documentation that accompanies the drive for detailed information about its disk geometry (cylinders, heads, sectors, and so on). You enter that information with Setup or the SCU.

See the documentation that accompanies the drive and your workstation installation guide for hardware installation instructions.

### Cards

When you add a card to a CTOS EISA/ISA system, you also need to configure it.

### Cards for ISA Workstations

ISA workstations can use only ISA cards. The CTOS ClientCard and CTOS VideoCard are set by default so that they do not conflict with each other. If you add an additional card to an ISA system conflict may occur. If so, the jumpers on that card need to be changed so that the settings do not conflict with those of the ClientCard and VideoCard. See your hardware installation guide for details.



### Cards for EISA Workstations

EISA workstations can use both EISA and ISA cards. When you first turn on a workstation that you have added cards to, the system reports that the hardware present does not match the configuration information in CMOS and prompts you to run the SCU. The system can see EISA cards, but shows slots that contain ISA cards as empty. Additional configuration is required for both types of cards.

The configuration information for the card is provided in a configuration file (.CFG file) that is supplied with the card. When you run the configuration utility it prompts you to mount the floppy disk that contains the configuration file.

Configuration files for the CTOS ClientCard, CTOS VideoCard, and CTOS IOPs are supplied with your system and with any extra cards that you order.

CTOS is set up to recognize the CTOS IOP card even if you do not run the SCU, however, to ensure that there are no system resource conflicts it is recommended that you run the SCU.

CTOS IOP Cards all use the same IRQ. They can share this IRQ with each other, but not with any different EISA or ISA cards that might be in the system.

### Video

EISA/ISA workstations support industry-standard video monitors, such as VGA (video graphics array). VGA monitors work with VGA controllers on the motherboard of your EISA/ISA workstation, or with a special CTOS VideoCard. See the *Software Release Announcement* for the Video Access Method for details on supported video hardware.

### CTOS VideoCard

Many CTOS applications have been designed to take advantage of the special features of CTOS video hardware. EISA/ISA workstations, however, do not include the video hardware that is necessary to fully support all CTOS video features. Therefore, a special ISA-bus card, called a CTOS VideoCard, is available to upgrade the video capabilities and provide full support for CTOS video.

### Standard VGA Display

If an EISA/ISA system is not equipped with a CTOS VideoCard, CTOS video attributes for applications such as OFIS Document Designer are displayed by standard VGA as follows:

<b>CTOS Video</b>	<b>Standard VGA</b>
Reverse video	Reverse video
Blinking	Blinking
Underline	Underline
Bold	Normal text
Struck-through	Normal text

Some of those attributes can be substituted for one another or changed to display in color. For example, you can choose to display either underlining or bold, but not both. If you choose to display bold, you can substitute a color in place of underlining.

See Appendix C in this manual for information about using the VAM VGA utilities for changing video attribute characteristics. See also the *CTOS Programming Guide* for detailed technical information about CTOS video methods.

### Keyboards and Mouse Devices

On an EISA/ISA workstation, you can use either a PC-style keyboard and mouse or an I-bus keyboard and mouse. (The I-bus keyboard and mouse are standard equipment on X-Bus workstations.)

**Note:** *The I-bus keyboard is for use with CTOS applications and CTOS/Vpc™ only; it is not supported for use with native DOS applications.*

The PC-style keyboard and mouse cables plug into small round connectors on the back of the workstation. Check the labeling in your hardware installation guide to determine correct connections.

The I-bus keyboard requires a special cable and an I-bus converter. The cable has a small, round plug on one end and a flat, click-in I-bus plug on the other end. The I-bus converter is a small box with a short cable on one end and a small round connector on the other end. To connect the I-bus converter, keyboard, and mouse, follow these steps:

1. Connect the short cable on the I-bus converter to the keyboard socket on the back of the workstation.
2. Plug the small round end of the keyboard cable into the I-bus converter.
3. Plug the flat click-in end of the keyboard cable into one of the I-bus connectors on the keyboard.
4. Plug the mouse into the other I-bus connector on the keyboard.

***Note:** Even when a mouse is properly connected to an EISA/ISA workstation, it is not recognized until after the Mouse Service has been installed. Therefore, it does not appear in the system configuration that is displayed when the workstation bootstraps.*





## Section 3

# Understanding System Software

## What Is System Software?

System software includes the operating system, system services, and configuration files that are required for a workstation or shared resource processor to function as a computer. This section describes the components of system software as they are installed on workstations and SRPs. See the *CTOS System Software Installation Guide* and the Software Release Announcements for installation instructions.

## Standard Software

Standard Software is a group of applications, utilities, and configuration files. It is distributed on floppy disks, QIC tape, and compact disc for installation on both workstations and SRPs. Many aspects of Standard Software are described in this manual. For detailed information about individual commands, see the *CTOS Executive Reference Manual*.

## Video Access Method

The Video Access Method (VAM) software contains video services that may be required in addition to the operating system. Because a number of different video modes are supported, including bit-mapped and VGA, those video services are packaged and installed separately from the operating system.

## Workstation Operating Systems

Workstation operating systems are divided into the following major categories:

CTOS I  
(real mode)

CTOS I is typically used on workstations containing 80186 CPUs. It can also be used, however, on certain models of 80286- and 80386-based workstations.

CTOS II  
(protected mode)

CTOS II is typically used on workstations containing 80286 and 80386. It takes advantage of the faster processing speed and extended memory of those workstations.

CTOS III  
(virtual memory)

CTOS III is used on processors containing 80386 and higher CPUs, such as EISA/ISA workstations and CTOS IOP cards. It allows multiple programs to be executed simultaneously in less memory than is required on CTOS II systems.

Virtual-memory operating systems are also protected-mode operating systems. Therefore, when used throughout this manual, the term *protected mode* includes virtual memory systems, unless otherwise specified.

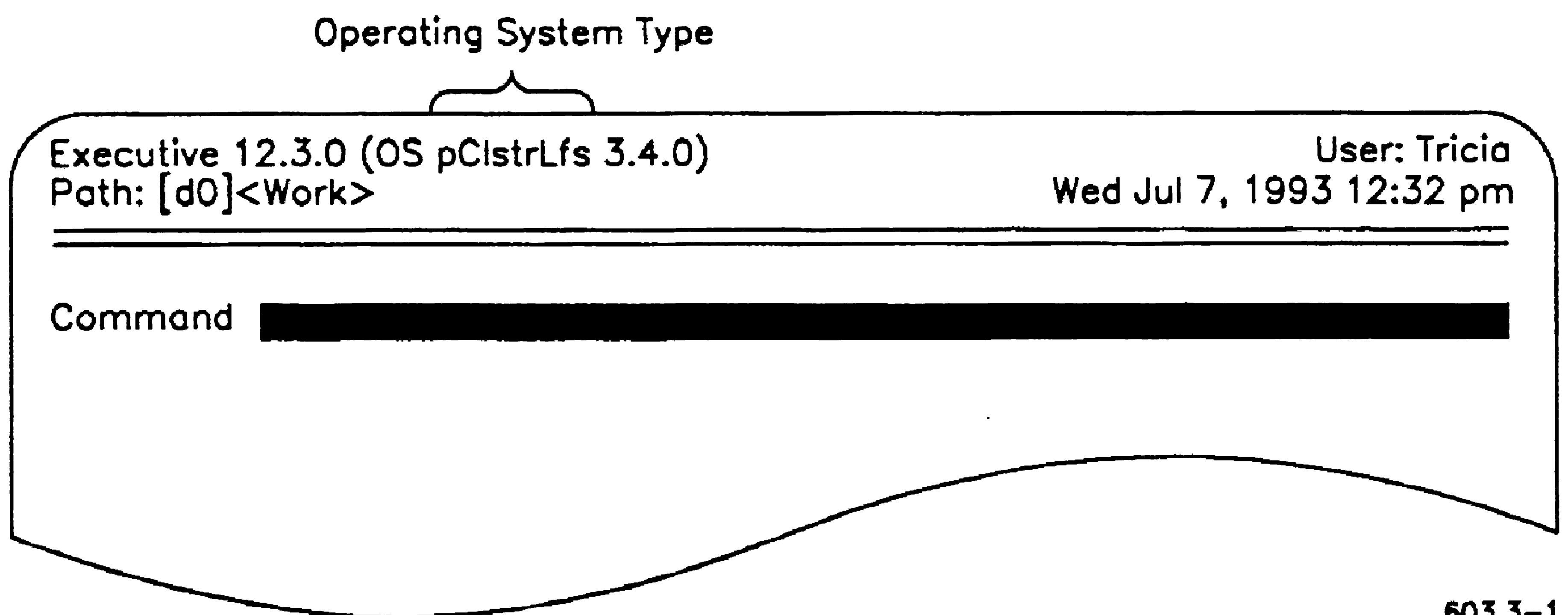
Beyond the main categories described above, the operating systems are divided into types, which determine whether a workstation functions as a server, a client, or a standalone system. Workstation operating system types are listed in Table 3-1 and described later in this section.

After system software has been installed and a workstation is up and running, the operating system type is displayed on the Executive screen, as shown in Figure 3-1.

Table 3-1. Workstation Operating Systems

Workstation or Processor Characteristic	Operating System Type		
	CTOS I (Real Mode)	CTOS II (Protected Mode)	CTOS III (Virtual Memory)
Server	NA	pSrvr	vSrvr
I/O processor card	NA	NA	vSrvr
Standalone	NA	NA	vStnd
Client with disks	t1ClstrLfs	pClstrLfs	vClstrLfs
Diskless client	t1Clstr	pClstr	vClstr
Client with disks (B27)	bAwsClstrLfs	NA	NA
Diskless client (B27)	bAwsClstr	NA	NA
Diskless client (B24)	v1Clstr	NA	NA

Figure 3-1. Operating System Identification



### Server Workstations

A server supports communications between itself and client workstations in the cluster. Any X-Bus or X-Bus+ workstation with a hard disk can function as a server when the appropriate operating system is installed. In addition, SuperGen Series 4000, 6000, and 7000 workstations can function as servers.

When possible, a workstation with an 80386 processor or higher is the best choice for a workstation server. The server controls access to many cluster-wide system services, which are described in Section 9, "Installing System Services." These can consume a great deal of memory, so the extended memory capabilities and faster processing speed of 80386 and 80486 hardware, in combination with a protected-mode or virtual memory operating system, are often essential.

### Client Workstations With Local File Systems

The term *local file system* (LFS) refers to a client workstation with a disk. Most LFS workstations are equipped with both a hard disk and a floppy disk drive; however, a workstation with a floppy disk drive only is considered to have a local file system. To function as an LFS, the workstation must be running the appropriate operating system, as shown in Table 3-1.

Local file systems can be used in different ways. For example, an LFS workstation that bootstraps from its own disk often functions as an independent system. If all applications and working files are stored locally, it is dependent on the server only for cluster-wide system services. Such a system usually remains operational when the server is not functioning.

Other workstations with local file systems may be used for storing working files only, while the operating system and application software reside on the server. In this case, applications are "downloaded" from a disk on the server to memory on the client workstation; data files, however, are stored on a local disk.

## Diskless Client Workstations

A diskless workstation does not contain a hard disk or a floppy drive. It boots from a disk on the server. Diskless workstations use a different operating system type than LFS workstations, as shown in Table 3-1.

## SRP Operating Systems

The SRP uses a set of operating systems to support its various processors. All types of its protected-mode processors run the same protected-mode operating system. Each type of real-mode processor, however, runs a slightly different real-mode operating system. The operating systems constantly communicate with one another to provide an integrated system. The set of SRP operating systems is named CTOS/XE.

Table 3-2 lists the prebuilt operating systems for SRP processors.

**Table 3-2. SRP Operating Systems**

Processor	Operating System	Mode
General Processor (including GP+SI, GP+CI, GP+EI)	pSrpGp.img	Protected
File Processor	rSrpFp.run	Real
Cluster Processor	rSrpCp.run	Real
Terminal Processor	rSrpTp.run	Real
Data Processor	rSrpDp.run	Real
Storage Processor	rSrpSp.run	Real





# Section 4

## Using Administrative Tools

### What Tools Are Available?

As a system administrator, you routinely use certain software utilities to monitor the status of the cluster, perform troubleshooting, and update system configuration files. The following software tools are commonly used by system administrators:

- The System Manager provides status information and access to frequently used commands.
- The Editor is used to customize configuration files for system software and other applications.
- Cluster View allows you to issue commands from a client workstation keyboard to a processor on the server.

An overview of these tools is presented in this section. More detailed instructions for using them to perform specific tasks are provided later in this manual.

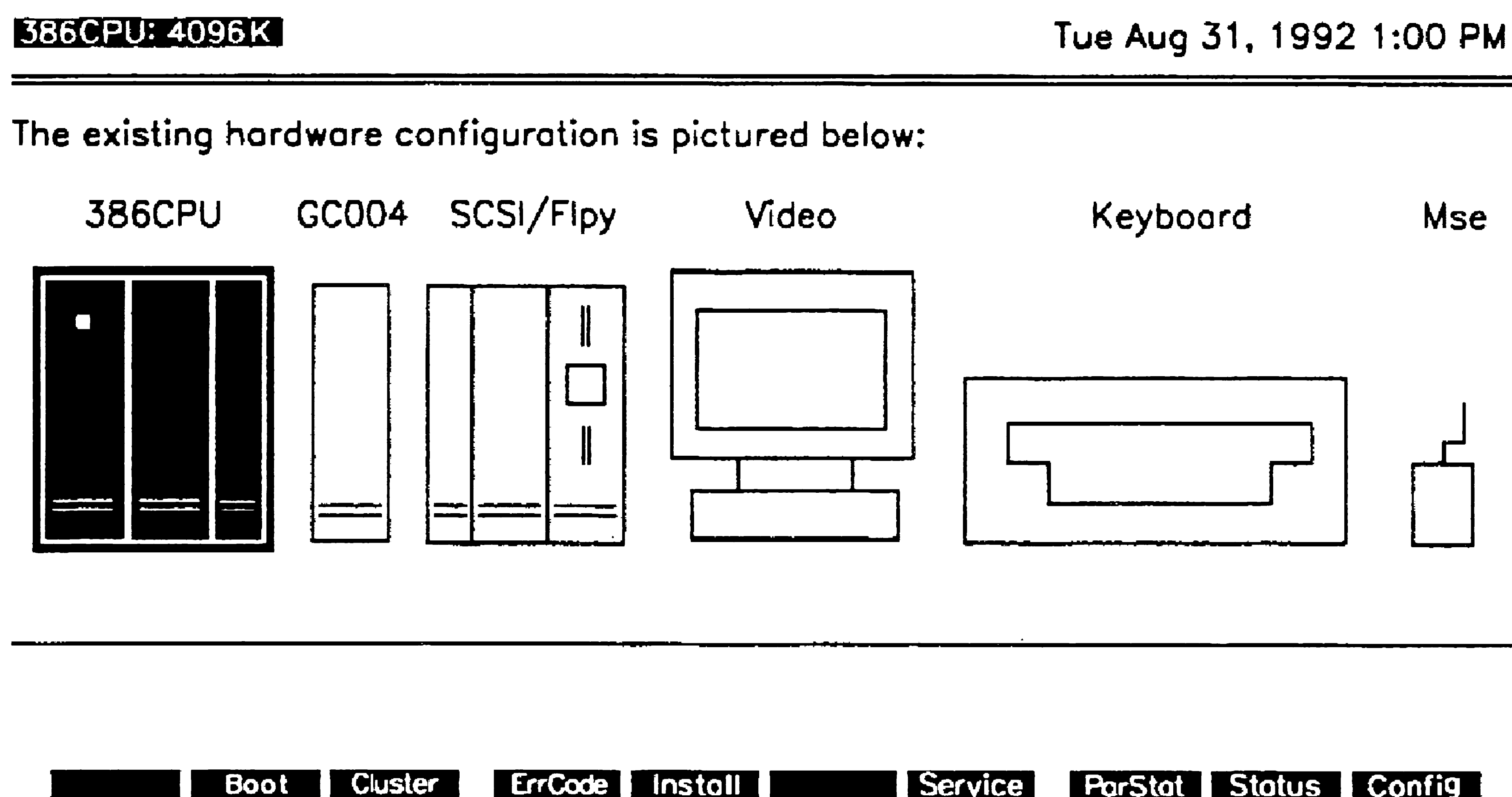
### System Manager

The System Manager is an administrative control center. It combines frequently used Executive commands into a single interface that you can use to perform the following types of tasks:

- Back up disks
- Format disks
- Edit configuration and command files
- Monitor memory, disk space, cluster, and network activity
- Install software

Figure 4-1 shows a sample System Manager display for a workstation.

Figure 4-1. System Manager Display on a Workstation



603.4-1

On a multiprocessor workstation, the System Manager displays the master processor and IOPs side by side, with slot numbers listed underneath. The slot number of the master processor is 9, and slot numbers for IOPs match the EISA slot numbers labeled on the back of the workstation. (You may need to know particular slot numbers to interpret error codes when you view the system log file with the **PLog** command.) Other components appear beside the processor icons and are described in the status area of the screen as you position the highlight.

### Starting the System Manager

To start the System Manager, follow these steps:

1. On the Executive command line, type **System Manager**.
2. Press **GO**.



The System Manager display is divided into the following sections:

Status area	The status area is located at the top of the screen, above the double bar. It displays information about the selected module and the system date and time.
Hardware components	Workstation modules and other hardware components are displayed and labeled in the center of the screen. The selected module is highlighted.
Function key menu	A function key menu is displayed at the bottom of the screen. You use function keys to invoke commands that pertain to the highlighted module. Function key names change as you select different modules.

### Using the System Manager on a Workstation

To use the System Manager, you select a hardware component and a function key for the task you want to perform. You can use the keyboard or the mouse to make selections from the System Manager display. Both methods are described below.

To exit the System Manager, press **FINISH**.

#### With the Keyboard

To make selections from the keyboard, follow these steps:

1. Use the arrow keys to position the highlight on the module you want to select.
2. Press the function key for the operation you want to select.

#### With the Mouse

To make selections with a mouse, follow these steps.

1. Move the mouse to position the highlight on the module you want to select.
2. Click the **Mark** button.

## Using Administrative Tools

---

3. Position the mouse cursor (usually an arrow) on the function key you want to select.
4. Click the **Mark** button.

### Function Keys

Each hardware component has its own set of functions. Therefore, function key labels change as you select different modules. See the **System Manager** command in the *CTOS Executive Reference Manual* for a complete list of function keys.

To return to the System Manager display after pressing a function key, press **CANCEL**.

The **F1 (Remote)** function key, which appears on some displays, starts the System Manager on an SRP server; it is described later in this section.

No function keys are provided for the keyboard, the Multiline Port Expander, or the PC Emulator.

### Using the System Manager on an SRP

You can use the System Manager on your SRP server in much the same way you use it on a workstation.

If the **F1 (Remote)** function key appears on your workstation display, press it to start the System Manager on the SRP server. In a few moments, the SRP System Manager is displayed, as shown in Figure 4-2.

While the System Manager is running on the SRP, the **F1** function key is labeled "Local." Press it to exit System Manager on the SRP and return to the workstation display.

*Note: If your workstation display does not show the **F1 (Remote)** key, see "Starting the System Manager With Cluster View," below.*

### Starting the System Manager With Cluster View

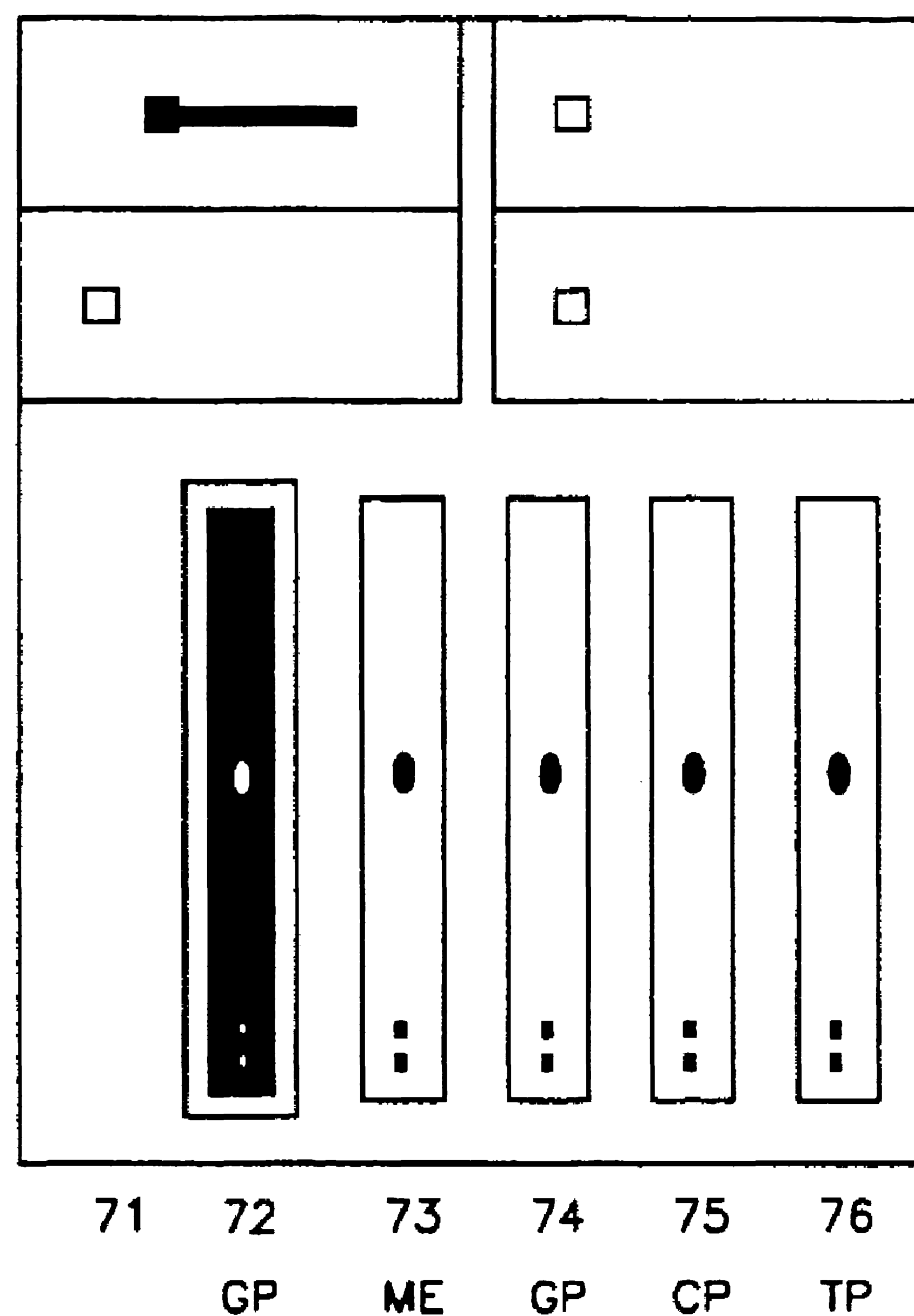
If the **F1** function key does not appear on the workstation display, you first start a Cluster View session and then execute the **System Manager** command. See "Cluster View," later in this section, for information about starting Cluster View on an SRP.

## System Manager Display on the SRP

On an SRP, the System Manager display resembles the system's hardware configuration, as viewed from the rear. A sample is shown in Figure 4-2.

Use the cursor keys, as described earlier in this section, to select a processor, disk, or tape drive. When you select a disk or tape drive, the processor that controls it becomes outlined in bold. If the SRP consists of more than three cabinets, press **NEXT PAGE** to display more cabinets. Press **PREV PAGE** to return to the picture of the cabinets previously displayed.

Figure 4-2. System Manager Display on an SRP



603.4-2

# The Editor

The Editor is a text-editing application, which system administrators use to edit configuration files, such as *Config.sys* and *SysInit.jcl*. The Editor is also a powerful programming tool for software developers.

This section provides basic instructions about editing and saving configuration files, which is sufficient for most system administrative functions. For more detailed information about the Editor, see the *CTOS Editor User's Guide*.

## Starting and Exiting the Editor

To start the Editor and open a file, follow these steps:

1. On the Executive command line, type **Editor**; then press **RETURN**.
2. Type the file specification of the file you want to open, as shown in the following example:

```
[File name(s)]  [Sys]<Sys>SysInit.jcl
```

3. Press **GO**.

To exit the Editor and save changes, press **FINISH**; then press **GO**.

## Editing Files

When you open a file with the Editor, the cursor appears beneath the first character of the file. To edit the file, you position the cursor, delete existing text, and then type new text. The keys you use to move the cursor and make corrections are described in Table 4-1.

Other useful keys include **F1** to redo the previous edit, **F2** to undo the previous edit, and **OVERTYPE** to type over existing text.

Table 4-1. Editor Application Keystrokes

Keystroke	Description
BACKSPACE	Deletes the character to the left of the cursor.
CODE+B	Moves the cursor to the beginning of the file.
CODE+DOWN ARROW	Moves the cursor to the bottom of the screen.
CODE+E	Moves the cursor to the end of the file.
CODE+LEFT ARROW	Moves the cursor to the beginning of the line.
CODE+RIGHT ARROW	Moves the cursor to the end of the line.
CODE+UP ARROW	Moves the cursor to the top of the screen.
DELETE	Deletes the character where the cursor is positioned.
DOWN ARROW	Moves the cursor down one line.
LEFT ARROW	Moves the cursor one character to the left.
RIGHT ARROW	Moves the cursor one character to the right.
SHIFT+RIGHT ARROW	Moves the cursor five characters to the right.
SHIFT+LEFT ARROW	Moves the cursor five characters to the left.
SHIFT+UP ARROW	Moves the cursor up three lines.
SHIFT+DOWN ARROW	Moves the cursor down three lines.
UP ARROW	Moves the cursor up one line.

## Opening, Closing, and Saving Files

To open a file, press **CODE+O**. Up to seven files can be opened on the screen simultaneously.

To open another window within a file, press **CODE+D**. To close a window, press **CODE+C**.

To clear a file from the screen, press **CODE+C**. This operation does not close or save the file.

To close and save all open files without exiting the Editor, press **CODE+S**.



### Searching for Text

The Search feature allows you to find a string of characters within a file. During a search, the Editor scans the file, starting from the current cursor position, and searches to the end of the file.

To perform a search, follow these steps:

1. Press **F6** to display the *Search* form.
2. Type the string of characters for which you want to search.
3. Press **GO**.

The first occurrence of the specified string, if present, is located and highlighted. To search for another occurrence, press **F6** and then **GO**.

### Replacing Text

The Search and Replace feature allows you to search for a requested string of characters and replace it with a new string. During a search and replace operation, the Editor starts at the beginning of the file and searches to the end of the file.

To search and replace, follow these steps:

1. Press **SHIFT+F6** to display the *Replace* form.
2. Fill in the form as shown in the following example:

```
Search for:  wildcard
Replace with: wild card
```

3. If you want the Editor to pause to allow you to confirm or deny each replacement, move the cursor to the *Confirm each?* field, and then press **Y**.

See the *CTOS Editor User's Guide* for information about the other options in this form.

4. Press **GO**.

### Manipulating Files

To copy, rename, and delete files, you use the Files menu. You also use the Files menu to set the default path or to display a list of files or the status of a volume.

To display the Files menu, press **CODE+F1**. To select an option, type the letter indicated. After selecting an option and filling in the form, press **GO** to activate the option. Menu options are described in Table 4-2.

**Table 4-2. Editor Files Menu Options**

Field Name	Description
<i>Open file</i>	Use this option to create or open a new or existing file, respectively. This is the same as pressing <b>CODE+O</b> .
<i>Delete file</i>	Use this option to delete a file or group of files. Fill in the file specification or wild card pattern of the file(s) you want to delete. If a file is currently displayed on the screen, its file specification is automatically filled into the form. To clear the field, press <b>CODE+DELETE</b> .
<i>Copy file</i>	Use this option to copy a file or group of files. Fill in the <i>Old file name</i> and <i>New file name</i> fields. If a file is currently displayed on the screen, its file specification is automatically filled into the <i>Old file name</i> field. To clear the field, press <b>CODE+DELETE</b> .
<i>Rename file</i>	Use this option to rename a file or group of files. Fill in the <i>Old file name</i> and <i>New file name</i> fields. If a file is currently displayed on the screen, its file specification is automatically filled into the <i>Old file name</i> field. To clear the field, press <b>CODE+DELETE</b> .
<i>Change default path</i>	Use this option to set the default volume and directory. Fill in the volume, directory, and other path entries you want to set.
<i>List files</i>	Use this option to list individual file names or groups of files according to the wild card pattern you specify. Fill in the form according to your needs. Press <b>CODE+C</b> to clear a list of files from the screen.
<i>Volume status</i>	This option lists the directories and free disk space on a volume. Fill in the volume, password, and node name. Press <b>CODE+C</b> to clear volume information from the screen.
<i>List open files</i>	Select this option to display a list of open files. Press <b>CODE+C</b> to clear the list from the screen.

# Cluster View

You use the Cluster View facility to issue commands from a client workstation keyboard to a processor on the server. Under ordinary conditions, commands are executed locally, on the workstation where you are typing. When you use Cluster View, however, your keyboard and monitor are communicating with a processor on the server. If your cluster is part of a BNet network, you can also use Cluster View on servers in the network.

## Using Cluster View on a Shared Resource Processor

An SRP does not have its own keyboard and monitor; therefore, to perform any work on the SRP, you must access its processors via Cluster View. For example, suppose that you want to perform a backup of an SRP disk, using its own tape drive. If you issue the **Volume Archive** command from your client workstation, data first travels to your client workstation for processing and is then sent back along the cluster line to the tape drive on the SRP. With Cluster View, however, only input and output take place on the workstation; the command is actually executed on an SRP processor, so the backup is completed more quickly.

## Using Cluster View on a Workstation Server

On a workstation server, you can use Cluster View to access processors on the server from a client workstation, or you can use Cluster View on the server itself to access IOP cards.

The use of Cluster View on single-processor workstation servers is optional but is often used for convenience or system security. It is particularly convenient to use Cluster View if your server is located in a remote computer room and you want to access it from your own office. In addition, if security is paramount, you can prevent others from using the server by removing its keyboard and monitor and then accessing it from a client workstation.



## Cluster View System Services

Cluster View consists of the following system services, which are installed on the server only:

- The Remote Keyboard Video Service provides basic Cluster View services.
- The Remote User Manager allows multiple Cluster View sessions to be started on servers running protected mode or virtual memory operating systems.

Both system services are described in more detail below. See also Section 9, “Installing System Services.”

### Remote Keyboard Video Service

The Remote Keyboard Video Service (RKVS) is required to use Cluster View. On SRP processors and CTOS I/O processors (IOPs), it is automatically installed as part of the bootstrap sequence. On single-processor workstation servers, because Cluster View is optional, you must make an entry in the system initialization JCL file to install it.

RKVS provides the basic Cluster View service. This allows one user at a time to execute commands on a processor via Cluster View. In addition, it runs in a “nonstop” mode; this means that after you initiate a Cluster View session, that session runs until the server is rebooted. For example, if you start a backup and then exit from Cluster View, the backup continues to run on the server. This can be convenient; however, it can be risky, too, because another user could gain access to that session and terminate the backup without your knowledge. For that reason, it is often wise to wait for a command to finish executing and then log out before you exit a Cluster View session.

### Remote User Manager

The Remote User Manager (RUM) enhances Cluster View services on processors running a protected-mode or virtual-memory operating system. RUM is an optional system service that you install during system initialization.

When RUM is installed, more than one Cluster View session can run on a processor simultaneously. This means that several users can execute commands or start applications via Cluster View at the same time. Also, when using RUM, each Cluster View session is terminated when the user exits Cluster View. This ensures that other users cannot gain access to a Cluster View session from another workstation.

If you are familiar with Context Manager, think of it to understand how the Remote User Manager works. With Context Manager on a workstation, you can run several applications at the same time. It keeps applications separate from one another, so that the work you perform in one partition is not affected by other partitions. RUM is similar, but it keeps applications separate on the server, for users who are executing them from multiple workstations within the cluster.

### Cluster View Commands

You start a Cluster View session from the Executive with either the **Administrator Cluster View** or **Cluster View** command. If the Remote User Manager is not installed, these commands are almost identical, so it doesn't matter which one you use. If the Remote User Manager is installed, however, these commands are used for different purposes, as described below:

#### Cluster View

Use this command to start multiple sessions on a processor running the Remote User Manager.

#### Administrator Cluster View

Use this command to prevent other users from starting a Cluster View session on a processor running the Remote User Manager. This ensures access to the primary partition, which is frequently necessary for administrative functions, such as installing system services. If another user attempts to start a session while you are using Administrator Cluster View, the following message appears:

```
Your session cannot be started.  An administrator
session is in progress.
```

## Installing the Cluster View Services

On multiprocessor-workstation IOPs and SRPs, RKVS is installed automatically when you boot the system; therefore, no separate installation is required to use it. RUM, however, is installed as a system service from the system initialization JCL file. For more information, see “Installing the Remote User Manager,” in Section 9, “Installing System Services.”

On single-processor workstation servers, RKVS is optional. Therefore, you install it as a system service from the system initialization JCL file. In addition, the Login Service and RUM are required if you want to continue using the single-processor server as a workstation. See “Using Cluster View on a Single-Processor Workstation,” later in this section.

## Starting Cluster View

The following procedure describes how to start a Cluster View session. Instructions for performing specific tasks with the **Cluster View** command are included throughout this manual. For more detailed information about the Cluster View command forms and parameter fields, see the *CTOS Executive Reference Manual*.

1. On the Executive command line, type **Cluster View** (or **Administrator Cluster View**); then press **RETURN**.

The following command form appears:

```
Cluster View
[Processor name] _____
[User name] _____
[User file password] _____
[Node name] _____
[Old XE run file?] _____
[Run file to invoke] _____
[Partition size] _____
```

2. Fill in the command form; fields are described in Table 4-3.
3. Press **GO**.

If RUM is not installed, the SignOn screen appears; sign on in the usual manner to start the Executive.

**Table 4-3. Cluster View Parameters**

Field Name	Description
<i>[Processor name]</i>	<p>Default: See below</p> <p>Enter the four-character processor ID of the processor you want to access. For client workstations, the default is the processor to which it is clustered. For multiprocessor workstations, there is no default.</p>
<i>[User name]</i>	<p>Default: SignOn user name</p> <p>Enter a user name that is valid for signing on to the server. (This means that the user configuration file must be present on the server.)</p>
<i>[User file password]</i>	<p>Default: SignOn password</p> <p>Enter a valid password for the server.</p>
<i>[Node name]</i>	<p>Default: Your own server</p> <p>Enter the name of the node with which you want to connect.</p>
<i>[Old XE run file?]</i>	<p>Default: No</p> <p>Enter <b>Yes</b> if the program you want to execute uses obsolete methods for writing to video. Use this option if screen output is garbled or nonexistent. This option does not provide workstation-quality video, but it does provide readable output.</p>
<i>[Run file to invoke]</i>	<p>Default: <i>[Sys]&lt;Sys&gt; Exec.run</i></p> <p>This field applies only to processors running the Remote User Manager. Enter the name of the run file you want to execute.</p>
<i>[Partition size]</i>	<p>Default: 400K bytes</p> <p>This field applies only to processors running the Remote User Manager. Enter the partition size, in K bytes, in which to execute the run file.</p>



## Working in a Cluster View Session

When you are working in a Cluster View session from a client workstation, your keyboard and monitor perform as though they were attached to the server, and the work you perform takes place on the server. Therefore, `[Sys]<Sys>` refers to the server's system volume; the exclamation point (!) is not required, because you are working on the server. While using Cluster View, you do not have access to your own local disks.

When you are working in a Cluster View session from the keyboard and monitor on a multiprocessor workstation, you have access to disks as usual. Disk access speed may be slower than when you are working on the workstation's master processor, however.

## Displaying the Cluster View Menu

While you are working in a Cluster View session, you perform certain functions with the Cluster View menu, as shown in Figure 4-3. To display the Cluster View menu, press the **HELP** key.

Figure 4-3. Cluster View Menu

Key	Action	Processor:GP00
FINISH	Terminate session	
CANCEL	Exit this menu	
HELP	Remote Help	
A	Debugger (Simple Mode)	
B	Debugger (Multi-Process Mode)	

603-4.3

While the Cluster View menu is displayed, the following keys function as described below:

<b>FINISH</b>	Terminates a Cluster View session.
<b>HELP</b>	Displays the Help facility for the application you are using.
<b>CANCEL</b>	Clears the Cluster View menu.
<b>A</b>	Starts the Debugger in Simple Mode (Administrator Cluster View sessions only).
<b>B</b>	Starts the Debugger in Multiprocess Mode (Administrator Cluster View sessions only).

### Exiting Cluster View

To exit Cluster View, follow these steps:

1. Press **HELP** to display the Cluster View menu (see Figure 4-3).
2. Press **FINISH**.

Control of the keyboard and monitor return to the client workstation from which the **Cluster View** command was executed.

### Using Cluster View on a Single-Processor Workstation

*Note: The concept of system initialization, as discussed below, has not been introduced yet in this manual. It is included in this section, however, to complete the documentation for Cluster View. See Section 9, "Installing System Services," for detailed information about system initialization and the use of JCL files.*

On single-processor workstation servers, RKVS is optional. Therefore, you install it as a system service from the system initialization JCL file. If you install RKVS only, the keyboard and monitor are not usable when a Cluster View session is in progress. This is appropriate if you want to limit the use of Cluster View to a single session, or if you plan to remove the keyboard from the server.

In other cases, though, you may want to retain use of the server as a workstation. To do so, you must install the Login Service and RUM, in addition to RKVS. The Login Service allows you to start a Cluster View session on the server workstation so that you can access it from its own keyboard.

### Installing RKVS, the Login Service, and RUM

The following system initialization JCL file shows entries for RKVS, the Login Service, and RUM in boldface type:

```
Job SysInit
Run [Sys]<Sys>RKVS.run
;Other system services are installed after RKVS but
;before the Login Service and RUM.
Run [Sys]<Sys>LoginService.run
RunNoWait [Sys]<Sys>RUM.run
End
```

The default partition size created by the Login Service is 500K bytes; however, it can be created larger or smaller by specifying its size, in K bytes, as shown in the following example:

```
Run [Sys]<Sys>LoginService.run, 800
```

The default number of RUM sessions that can be open simultaneously is 2; however, you can specify up to 13 sessions as a parameter to RKVS, as shown in the following example:

```
Run [Sys]<Sys>RKVS.run, 4
```

Although the parameter value is placed after RKVS, it has no effect if RUM is not subsequently installed.

### Using the Keyboard and Monitor

When the Remote User Manager is installed on a workstation server, the following message appears on the server's monitor:

```
The Remote User Manager is in use.
```

To use the server's keyboard and monitor, you must start a Cluster View session on the server. To do so, press **ACTION+NEXT** to display the SignOn screen; you can then sign on to the server in the usual manner.



You cannot use Context Manager when running a Cluster View session on the server. In addition, when a Cluster View session is running on the server, an administrator session cannot be started from anywhere within the cluster.

To exit a session on the server, follow these steps:

1. On the Executive command line, type **Logout**; then press **GO**.
2. If the SignOn screen is displayed, press **ACTION+FINISH**.

# Section 5

## Bootstrapping

### How a System Bootstraps

The term *bootstrap* (or just boot) is derived from the saying “to pull oneself up by his or her own bootstraps,” meaning without any help.

Each system contains a component called a *bootstrap ROM*, which contains the first program executed when the system is turned on or reset. (ROM stands for read-only memory.) Because this program is self-contained within the processor, the system is said to bootstrap, or to get itself started without any help. The instructions contained on the bootstrap ROM are permanently etched onto it when it is manufactured.

A major function of the bootstrap ROM is to locate and load the correct operating system (also called the *system image*). Therefore, the operating system must be stored in a file that is recognized by the bootstrap ROM. That file is named `<Sys>SysImage.sys`.

### Bootstrapping a Workstation

When a workstation boots from a system image located on its own disk, it is said to boot locally or from a local disk. When you turn on a workstation, the bootstrap ROM searches for a file named `<Sys>SysImage.sys`. It searches disks in a predetermined order, depending on the workstation model, and loads the first occurrence of `<Sys>SysImage.sys` containing a valid system image file. At that point, the disk containing the system image loaded by the bootstrap ROM becomes the system volume and is known as `[Sys]`.

If no disk on an X-Bus or X-Bus+ workstation contains a bootable system image, the bootstrap ROM attempts to boot from the server.

If no disk on an EISA/ISA workstation contains a bootable system image, the bootstrap ROM on the ClientCard attempts to boot from the server, if it has been properly configured. (See the hardware documentation for the ClientCard.)

The following steps describe the workstation bootstrap sequence:

1. The bootstrap ROM searches for the bootable system image file.
2. The bootstrap ROM loads the system image into memory and transfers control of the workstation to the operating system. Parameters in the operating system configuration file are implemented at this time.
3. The system initialization JCL file is executed.
4. The SignOn program is started.

*Note:* On EISA/ISA workstations, the default bootstrap sequence does not proceed past [f0] if an unbootable floppy disk is in the floppy drive. Remove the floppy disk before rebooting the workstation.

The default operating system configuration file is named `[Sys]<Sys>Config.sys` and is described in detail in Section 16, "Configuring Operating Systems."

The system initialization JCL file is named `[Sys]<Sys>SysInit.jcl` and is described in detail in Section 9, "Installing System Services."

See Section 18, "Customizing Standard Software," for information about the SignOn programs and the SignOn screen display.

## Bootstrapping I/O Processors

On a multiprocessor workstation, the master processor boots from the system image file in `<Sys>SysImage.sys`. The master processor then boots the I/O processors (IOPs), as described in the following steps:

1. The bootstrap ROM on the master processor loads the operating system for the master processor.
2. The master processor reads the operating system configuration file and implements configuration parameters for itself.
3. The master processor boots the IOPs and implements parameters, as defined in the operating system configuration file.
4. After the boot sequence, the master processor executes the system initialization JCL file to install system services on itself and the IOPs.
5. The master processor executes the SignOn program.

## Bootstrapping a Workstation From the Server

The following workstation configurations bootstrap from a system image located on the server:

- Client workstations without disks
- Client workstations with disks that are initialized for data storage only (that is, not containing an operating system file)

If the bootstrap ROM does not find the system image on a local device, it searches for it on the server. To locate the correct system image, the bootstrap ROM searches for a file named as follows:

`[Sys]<Sys>WsNNN>SysImage.sys`

where *NNN* is a three-digit workstation-type number, as listed in Table 5-1.

*Note:* For an EISA/ISA workstation to boot automatically from the server, its ClientCard must be properly configured. See the hardware documentation for the ClientCard.

When a workstation boots from `WsNNN>SysImage.sys`, a matching operating system configuration file (`WsNNN>Config.sys`) and system initialization file (`WsNNN>SysInit.jcl`) are executed if they exist. In addition, a SignOn text file (`WsNNN>SignOn.txt`) can be created for each workstation-type number; see Section 18, “Customizing Standard Software.”

## Workstation-Type Numbers

The workstation-type number (*NNN*) is derived from the following:

- A processor number etched onto the bootstrap ROM
- The devices detected by the bootstrap ROM while it attempts to bootstrap locally

Table 5-1 lists workstation-type numbers. The bootstrap ROM uses the workstation-type number (*NNN*) to locate the correct operating system on the server. For example, the bootstrap ROM on a diskless B38 workstation searches for `[Sys]<Sys>Ws242>SysImage.sys` on the server.

**Table 5-1. Workstation-Type Numbers**

<b>Workstation Number (NNN)</b>	<b>Processor</b>	<b>File System</b>
090	ClientCard	Hard disk
092	ClientCard	Diskless
125	B27	Hard disk(s)
126	B27	Floppy disk(s) only
127	B27	Diskless
200	B24	Diskless
210	B38* and 386 NGEN*	Hard disk(s)
211	B38* and 386 NGEN*	Floppy disk(s) only
212	B38* and 386 NGEN*	Diskless
213	Series 5000	Hard disk(s)
219	SG2100-320	Diskless
220	B39 and Series 386i	Hard disk(s)
230	Series 286i	Hard disk(s)
231	Series 286i	Floppy disk(s) only
240	B28, 286 NGEN, B38*, and 386 NGEN*	Hard disk(s)
241	B28, 286 NGEN, B38*, and 386 NGEN*	Floppy disk(s) only
242	B28, 286 NGEN, B38*, and 386 NGEN*	Diskless
250	B26 and 186 NGEN	Hard disk(s)
251	B26 and 186 NGEN	Floppy disk(s) only
252	B26, CWS, and 186 NGEN	Diskless

\*Boot ROM versions 3.2 and higher boot from 210, 211, or 212; versions lower than 3.2 boot from 240, 241, or 242. The boot ROM version is displayed on the Bootstrap menu; see "Using the Bootstrap Menu," later in this section.



## Using the Bootstrap Menu

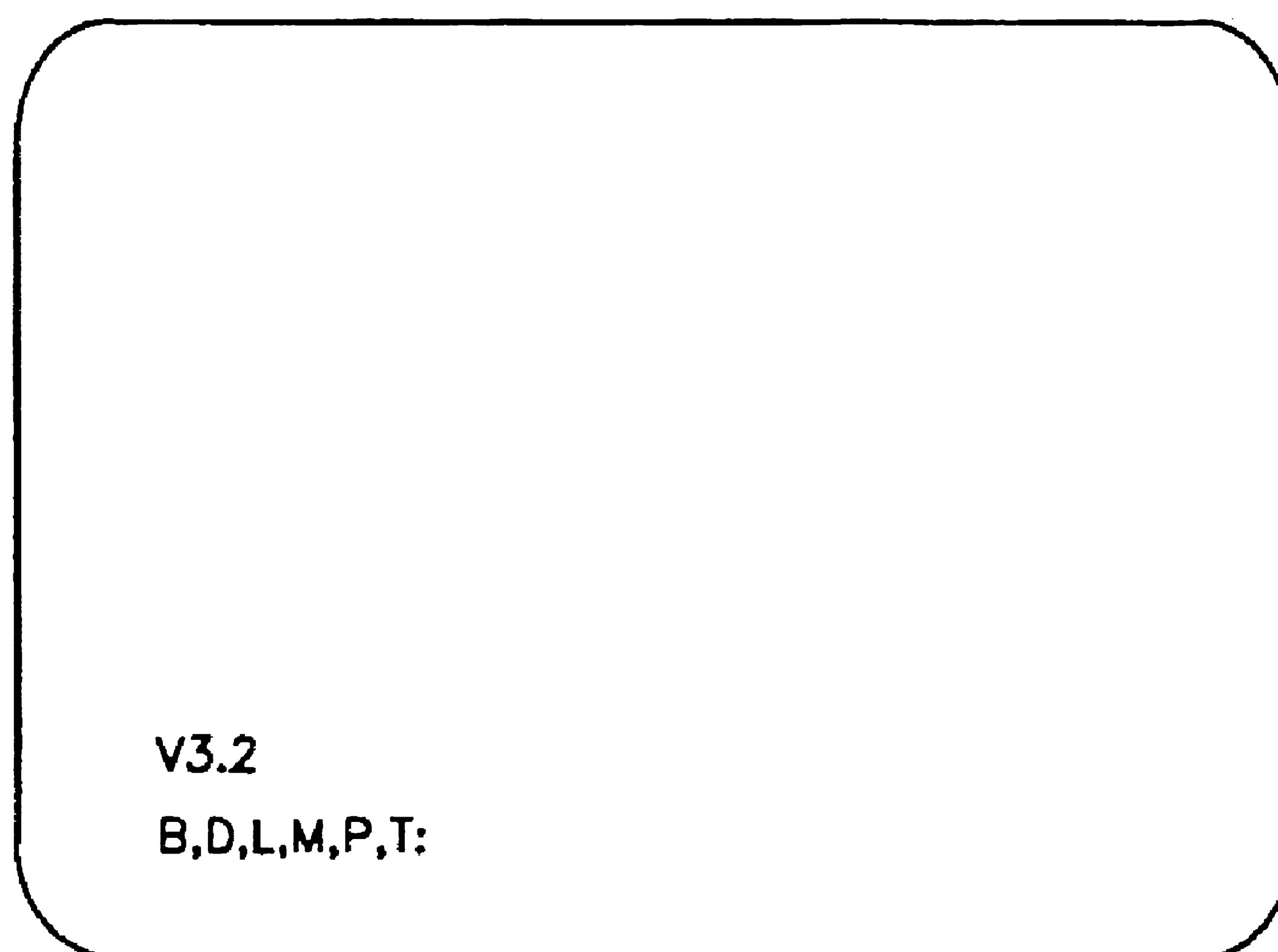
You can force a client workstation to boot from the server rather than from its own disk. On some workstations, you can also bootstrap a workstation-type number that is not recognized by the bootstrap ROM. To do so, you manually initiate the bootstrap sequence from the Bootstrap menu.

A representative Bootstrap menu is pictured in Figure 5-1. The bootstrap ROM version number is displayed above the Bootstrap menu. The menu options appear as single characters below the version number. Several commonly used options are described in this section.

*Note: Options on the Bootstrap menu vary among different workstation models. See your hardware diagnostics manual for a complete description.*

Two procedures for using the Bootstrap menu, one for X-Bus and X-Bus+ workstations and another for EISA/ISA workstation, are included below.

**Figure 5-1. Bootstrap Menu**



603.5-1

### Caution

The following procedure interrupts applications that are currently running on the system and may cause loss of data. Therefore, always exit applications before bootstrapping.

### Procedure for X-Bus and X-Bus+ Workstations

To bootstrap an X-Bus or X-Bus+ workstation from the server using a specific workstation-type number, follow these steps:

1. To display the Bootstrap menu (see Figure 5-1), hold down the space bar while you turn on or reset the workstation.
2. Release the space bar when the Bootstrap menu appears.
3. Type **t** (either uppercase or lowercase).
4. Type the workstation-type number, for example, 240.
5. Press **RETURN**.
6. Type **b**.

### Procedure for EISA/ISA Workstations

To bootstrap an EISA/ISA workstation from the server (except for notebooks) using a specific workstation-type number, follow these steps:

1. Press the Reset or On/Off switch button.
2. If prompted, press **F1**.
3. After you are informed that the BIOS installed successfully, press the space bar to display the Bootstrap menu. (You do not need to hold it down.)

*Note: The boot process does not stop after the BIOS is installed, so you will need to watch the screen.*

4. Type **t** (either uppercase or lowercase).
5. Type the workstation-type number, for example, 090.
6. Press **RETURN**.
7. Type **c**.



## Indirect Bootstrapping

If your server is running CTOS II or CTOS III, you do not need to store multiple copies of the same system image in different files for different workstation-type numbers. Instead, each file named *[Sys]<Sys>WsNNN>SysImage.sys* can contain the file specification of the actual system image file.

For example, when you install the CTOS II operating systems, the file named *[Sys]<Sys>Ws240>SysImage.sys* contains the following file specification, which acts as a pointer to the actual system image file:

*[Sys]<Sys>pClstrLfs.img*

## Workstation Hardware IDs

On workstations equipped with appropriate hardware, you can assign a hardware ID number (*HwNNN*) and use it instead of a workstation-type number (*WsNNN*) in the system initialization file specification, as shown below:

*[Sys]<Sys>HwNNN>SysInit.jcl*

where *NNN* is the hardware ID number assigned to the workstation.

In addition, you can create a customized SignOn text file (*HwNNN>SignOn.txt*) for individual hardware ID numbers. See Section 18, "Customizing Standard Software," for more detailed information.

On workstations running CTOS III, this feature also exists for the operating system configuration file (*HwNNN>Config.sys*).

After a workstation bootstraps from the server, the operating system reads the workstation's hardware ID. If the files listed above exist for that hardware ID, they take precedence over the corresponding workstation-type number files.

You assign a hardware ID number with the **Write Hardware ID** command. You can find out whether a hardware ID number is currently assigned with the **Read Hardware ID** command. See the *CTOS Executive Reference Manual* for information about using those commands.

### Using the Bootstrap Command

You can bootstrap a workstation from any valid system image file by using the **Bootstrap** command. In addition, on CTOS III workstations only, you can specify an operating system configuration file other than the default, as shown in the following example:

```
Bootstrap  
[File name]           [Sys]<Test>TestOS.img  
[Sys volume or WsNNN]  
[OS configuration file] [Sys]<Test>TestConfig.sys
```

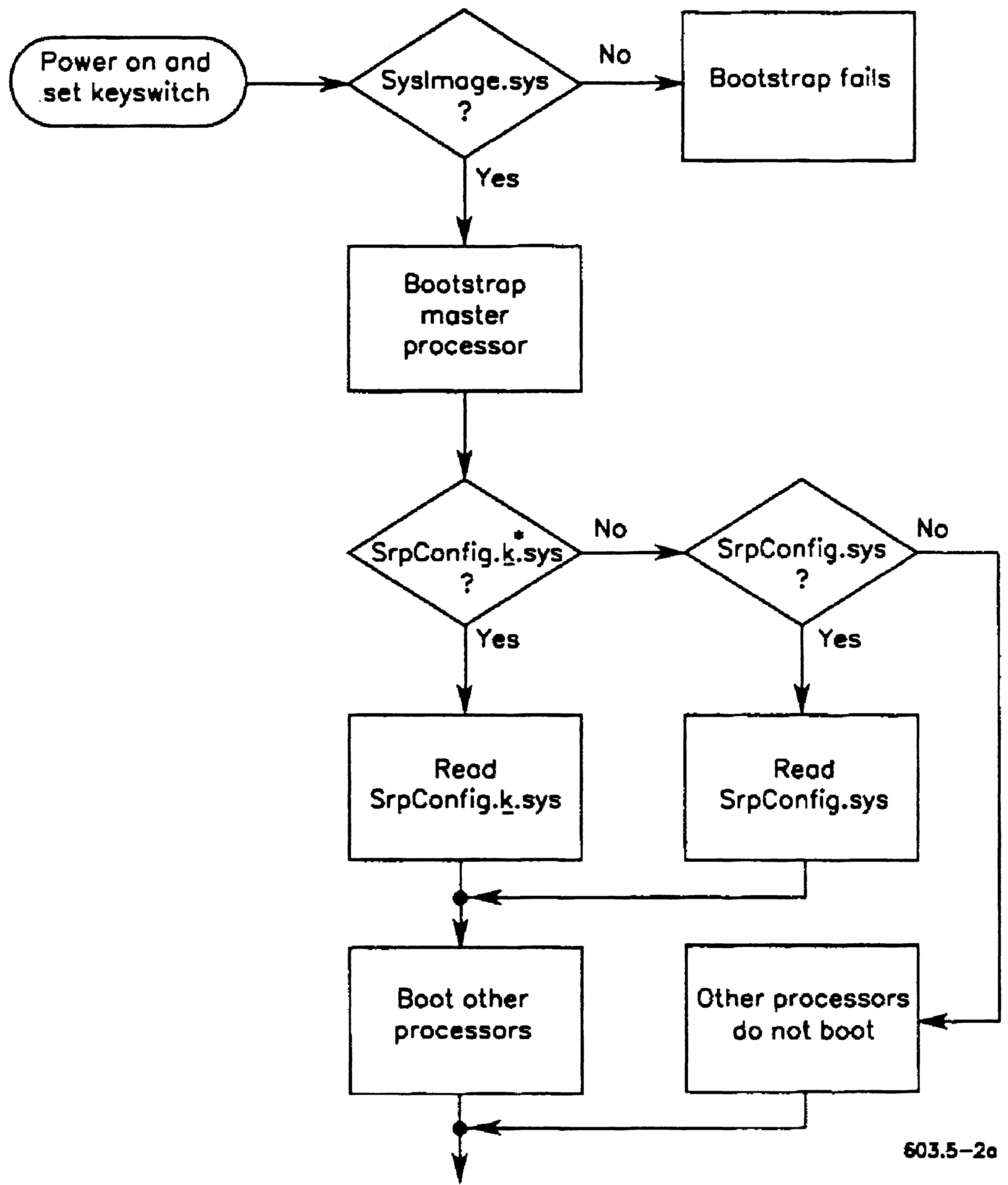
See the *CTOS Executive Reference Manual* for more detailed information about the **Bootstrap** command.

### Bootstrapping a Shared Resource Processor

On a shared resource processor, the master processor bootstraps from the system image file, *[Sys]<Sys>SysImage.sys*. The master processor then boots the system's other processors. The following steps describe the SRP bootstrap sequence and are illustrated in Figure 5-2.

1. The bootstrap ROM on the master processor loads the operating system for the master processor.
2. The master processor reads the operating system configuration file and implements configuration parameters for itself.
3. The master processor boots the other processors and implements parameters, as defined in the operating system configuration file.
4. After the boot sequence, the master processor executes the system initialization JCL file to install system services.

Figure 5-2. SRP Bootstrap Sequence

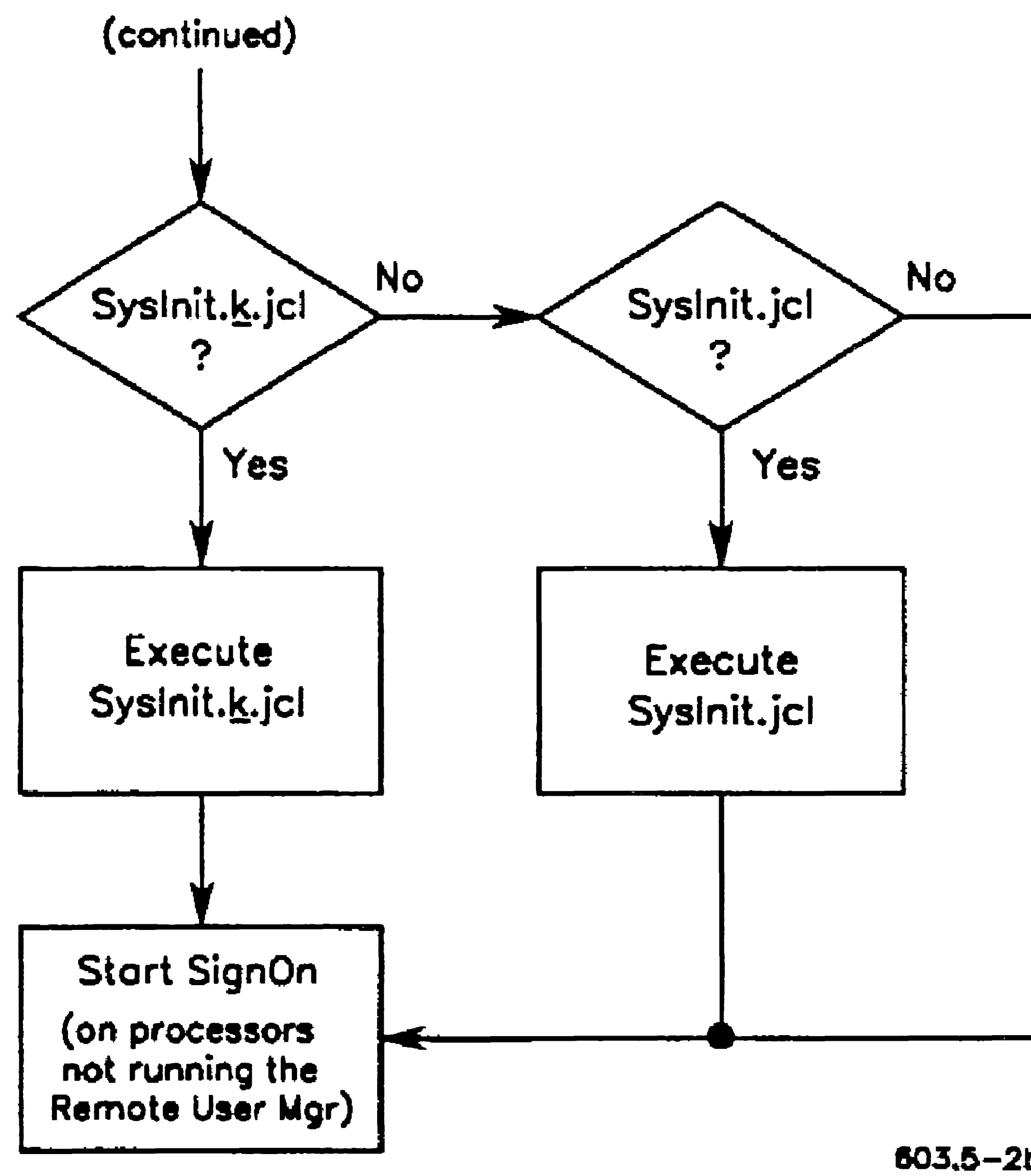


603.5-2a

(continued on next page)

\* k represents keyswitch position N, M, or R.

Figure 5-2. SRP Bootstrap Sequence (cont.)



## Keyswitch Positions

To bootstrap an SRP, you turn the keyswitch on the front panel (see Figure 2-2) to one of three positions, as described below:

- N        Stands for the *normal* keyswitch position; it is frequently used for normal working conditions. In this position, the Reset button is disabled to prevent rebooting by unauthorized persons, and the SRP reboots automatically after a system crash.
  
- M        Stands for the *manual* keyswitch position; it is most often used in software development environments where it is necessary to debug processor crashes. In this position, the Reset button is enabled and the system does not reboot itself after a processor crash.
  
- R        Stands for the *remote* keyswitch position; it is used to load diagnostics programs on the master processor. It can also be used to boot a minimum configuration for troubleshooting. In this position, the Reset button is disabled and the system automatically reboots after a system crash.

## Keyswitch-Controlled Files

In addition to the differences inherent in the keyswitch positions, these positions control loading and execution of the operating system configuration file (*SrpConfig.k.sys*) and the system initialization JCL file (*SysInit.k.jcl*). If either of those files exists with *k* corresponding to the keyswitch position from which booting takes place, that file is used. See Figure 5-2 for the exact sequence.

See Section 16, “Configuring Operating Systems,” for information about the operating system configuration file. See Section 9, “Installing System Services,” for information about the system initialization JCL file.



# Section 6

## Implementing System Security

### How Passwords Work

The CTOS operating system provides a multilevel approach to password protection. As system administrator, you assign passwords and protection levels to volumes, directories, and files. The correct password is then required to gain access to files or to execute certain commands on the system. Passwords are frequently used to protect the system in the following ways:

- To protect a volume from commands that destroy data
- To limit access to certain directories or files
- To restrict access to the entire system

### Protecting Volumes

A *volume password* is the key to system security. Without a volume password, the password protection mechanism is not activated. Therefore, you must assign a volume password to each disk that you want to protect.

The volume password is the “master” password for the disk. It gives the user unrestricted access to the volume. It must be supplied to perform the following operations.

- To reinitialize the volume
- To change the name or password of the volume
- To back up a volume



Volume protection alone, however, does not restrict access to directories and files on the disk. To do this, you must implement additional levels of system security, which are described later in this section.

### Assigning a Password to the System Volume

When you install system software, the system volume is formatted with the **Configure Disk** command as part of the installation procedure. At that time, you are prompted to enter parameter values for the system volume, such as a volume name and password.

#### Caution

The **Configure Disk** and **Format Disk** commands destroy all data on the disk. Do not use them to assign a password to a disk containing data. Instead, see "Changing a Volume Password," later in this section.

### Assigning Passwords to Other Volumes

After you have installed system software, you initialize the other disks on the system (see Section 11, "Adding Hard Disks"). You can assign the same password to all volumes on a system, or you can make each volume password unique. If you, the system administrator, are the only user with access to volume passwords, your job will be simplified by using the same password on all disks. If, however, other users need volume-level access to certain disks, you may want to assign a different password to each disk.

To assign a volume password when you initialize a disk, enter a password of twelve or fewer characters in the *[New volume password]* field of the **Format Disk** (or **Configure Disk**) command form. The password may contain letters, numbers, and punctuation, as shown in the following example:

```

Format Disk
Device name                d0
[Device password]         ##
[Current volume password] #####
[New volume name]         Freddie0
[New volume password]     J-123
[Configuration file]
[Format template]
[Device template]
[Print file]
[Overwrite ok?]
[Bad spot file]
[Recalculate defaults?]
[CTOS partition size in MB]
  
```

Be sure to keep a record of the passwords you assign. After the disk is initialized, you are not able to view the volume password on the screen. See also "Initializing an Unformatted Disk," in Section 11, "Adding Hard Disks."

**Note:** *Device passwords are assigned to hard disk drives by the operating system. For the prebuilt operating systems, device passwords match device names, that is, d0, d1, and so on.*

### Changing a Volume Password

#### Caution

Some applications, such as electronic mail, require reconfiguration after you change a volume name or password.

You can change or assign a volume password with the **Change Volume Name** command. To do so, follow these steps:

1. On the Executive command line, type **Change Volume Name**; then press **RETURN**.
2. Fill in the command form as shown in the following example. Parameter fields are described in Table 6-1.

Change Volume Name

Device name	d0
[Device password]	
[Old volume password]	####
New volume name	NewVol
[New volume password]	efgh

3. Press **GO**.

**Table 6-1. Change Volume Name Parameters**

Field Name	Description
<i>Device name</i>	Enter the device name.
<i>[Device password]</i>	Default: None  Enter the device password for the disk. (In most cases you can leave this field blank; it is required only if the disk is not a valid volume.)
<i>[Old volume password]</i>	Default: Default password  Enter the password currently assigned to the volume.
<i>New volume name</i>	Enter the volume name you want to assign to the disk. If you are changing the volume password only, enter the <i>current volume name</i> (see the caution above).
<i>[New volume password]</i>	Default: Currently assigned password  Enter the new password you want to assign to the volume.

## Protecting Directories

When you create a directory, you can assign a directory password and set a protection level. The protection level determines whether a volume, directory, or file password is required to gain access to a file, and it is assigned to every file that is subsequently created in the directory.

Protection levels are listed in Table 6-2. Protection levels also determine the type of access that is permitted to a file, as described below:

- *Read access* allows the user to view files or load programs. Read access is required for the operating system, configuration files, and application programs.
- *Modify access* allows the user to make changes to a file. Modify access is required to create and make changes to files with applications, such as OFIS Document Designer™ or OFIS Graphics, and to install or update software applications.

Some examples for using different protection levels are included later in this section.

**Table 6-2. Protection and Access Levels**

Protection Level	Password Level Required		Description
	To Read	To Modify	
15	None	None	No password is required to read or modify the file unless the files has a password.
5	None	Volume Directory	No password is required to read the file. A volume or directory password is required to modify the file.
0	Volume Directory	Volume Directory	A volume or directory password is required to read or modify the file.
7	None	Volume Directory File	No password is required to read the file. A volume, directory, or file password is required to modify the file.
3	Volume Directory File	Volume Directory File	A volume, directory, or file password is required to read or modify the file.
1	Volume Directory File	Volume Directory	The file can be read with a volume, directory, or file password, but a volume or directory password is required to modify it.
23	None	Volume File	No password is required to read the file, but a volume or file password is required to modify it.
19	Volume Directory File	Volume File	The file can be read with a volume, directory, or file password, but a volume or file password is required to modify it.
51	Volume File	Volume File	A volume or file password is required to read or modify the file.

## Assigning a Password to a Directory

You can assign a directory password when you create the directory or later with the **Set Directory Protection** command. To assign a password when you create a directory, follow these steps:

1. On the Executive command line, type **Create Directory**; then press **RETURN**.
2. Fill in the command form as shown in the following example. Parameter fields are described in Table 6-3.

```

Create Directory
New directory name(s)           NewDir
[Default protection level (15)] 0
[Maximum number of files (75)]
[Password for new directory]    TFS
[Volume password]               #####
    
```

3. Press **GO**.

See the **Create Directory** command in the *CTOS Executive Reference Manual* for more information.

**Table 6-3. Create Directory Parameters**

Field Name	Description
<i>New directory name(s)</i>	Enter a name for the directory you want to create.
<i>[Default protection level (15)]</i>	Default: 15 Enter the protection level you want assigned to new files when they are created (see Table 6-2).
<i>[Maximum number of files (75)]</i>	Default: 75 Enter the maximum number of files that can be created in the directory. Note that after a directory has been created, its size cannot be changed.
<i>[Password for new directory]</i>	Default: Active password Enter the password you want to assign to the directory.
<i>[Volume password]</i>	Default: Active password Enter the volume password for the volume on which you want to create the directory.



## Changing a Directory Password

To change or assign a password to a directory that already exists, follow these steps:

1. On the Executive command line, type **Set Directory Protection**; then press **RETURN**.
2. Fill in the command form as shown in the following example. Parameter fields are described in Table 6-4.

```
Set Directory Protection
Directory name(s)                NewDir
[Volume or directory password]   ###
[New file protection level (current)] 5
[New directory password]         RAC
```

3. Press **GO**.

This command does not affect files already stored in the directory. See "Protecting Files," later in this section, to learn how to change the password and protection level of a file.

**Table 6-4. Set Directory Protection Parameters**

Field Name	Description
<i>Directory name(s)</i>	Enter the name of the directory that has the password and/or protection level you want to change.
<i>[Volume or directory password]</i>	Default: Current default password Enter the currently assigned directory password or the volume password.
<i>[New file protection level (current)]</i>	Default: Currently assigned protection level Enter the protection level you want to assign to the directory. If you leave this field blank, the protection level is not changed.
<i>[New directory password]</i>	Default: Currently assigned password Enter a new password for the directory. If you leave this field blank, the password is not changed.



## Protecting the <Sys> Directory

Protection level 5 is an appropriate choice for the <Sys> directory. It allows read access but does not allow modification without a password. If you “overprotect” <Sys> (for example, with a protection level of 0), a password is required to read any file, even files that are required to bootstrap the workstation.

To assign a password and protection level to the <Sys> directory, use the **Set Directory Protection** command, as shown in the following example:

```

Set Directory Protection
Directory name(s)           Sys
[Volume or directory password]  #####
[New file protection level (current)]  5
[New directory password]      KeepOut
    
```

After you assign a password to <Sys>, you may also need to set the protection level of the files that are stored there. See “Protecting Files,” later in this section.

## Limiting Access to Directories

If several users share disks on the server, you may want to create an individual directory with a unique password for each user. When you create the directories, assign a protection level of 0 or 3 (see Table 6-2); this prevents access by users who do not know the correct password for the directory. Of course, if the volume password is supplied, a user can access any file or directory on the disk.

## Protecting Files

File passwords and protection levels are frequently used to accomplish the following:

- To protect groups of files, such as those in the <Sys> directory, from being modified by unauthorized users
- To protect individual files from being either read or modified by unauthorized users

Both situations are described below.

### Assigning a Protection Level to a Group of Files

The password and protection level of existing files do not change when you change the protection level of the directory in which they are stored. You assign the protection level to individual files in a separate step, as described in the following procedure.

1. On the Executive command line, type **Set Protection**; then press **RETURN**.
2. Fill in the command form as shown in the following example. Parameter fields are described in Table 6-5.

```
Set Protection
File list           [Sys]<Sys>*
New file protection level 5
[New password]
[Confirm each?]
```

3. Press **GO**.

**Table 6-5. Set Protection Parameters**

Field Name	Description
<i>File list</i>	Enter the list of files that have protection levels and/or passwords you want to change.
<i>New file protection level</i>	Default: Currently assigned protection level Enter the protection level you want to assign to the files. If you leave this field blank, the protection level is not changed.
<i>[New password]</i>	Default: Currently assigned password Enter a new password for the files. If you leave this field blank, the password is not changed.
<i>[Confirm each?]</i>	Default: No If you enter <b>Yes</b> , you are prompted for confirmation before the protection level and/or password is changed for each file. If you enter <b>No</b> or leave this field blank, you are not prompted for confirmation.

## Assigning a Unique Password to a File

When you create a new file, it automatically inherits the password and protection level of the directory in which it is created. You can, however, assign a unique password or change its protection level with the **Set Protection** command.

The following example shows how to assign a password to a file. In addition, by assigning protection level 23, you allow users read access without a password, but require them to enter the file password to modify the file. To truly protect a file, assign a password that is different from the directory password.

```

Set Protection
File list                [d1]<Work>Read.doc
New file protection level 23
[New password]           secret
[Confirm each?]
    
```

To require a password for both read and modify access, assign protection level 51 and a unique password. This method is frequently used to assign passwords to user configuration files. See “Restricting Access to the System,” below.

## Restricting Access to the System

You can use file passwords to prevent users from signing on without a password. To do this, you assign a password and protection level 51 to each user configuration file (see “Assigning Passwords to User Names,” in Section 7, “Customizing User Environments”). You can do this with the **User File Editor** command when you create the user file, or you can use the **Set Protection** command, as shown in the following example:

```

Set Protection
File list                [Sys]<Sys>Freddie.user
New file protection level 51
[New password]           popcorn
[Confirm each?]
    
```

When you protect a user file, the user must supply the file password or the volume password to sign on to the workstation.

### Allowing Access to a Single Directory

You can also implement the following protection scheme, which requires a valid password for signing on and allows access to files in one directory only:

1. Create a directory with protection level 0 and a unique password.
2. Use the **Set Protection** command to assign protection level 51 and the same password you assigned to the directory to the user configuration file.
3. Within the user configuration file, specify the same password again in the *:SignOnPassword:* field (see “SignOn Options,” in Section 7, “Customizing User Environments”).

With the above method, a single password functions as both the user-file password and the directory password for the user's directory. When the password is entered in the SignOn form, it provides access to the user configuration file; when it is read again from the user configuration file, it provides access to the user's directory.

Alternatively, you could assign different passwords to both the user file and the directory. The *:SignOnPassword:* entry in the user file then overrides the password that the user enters in the SignOn form. This allows the user access to the directory without knowing its directory password.

### Eliminating Known User Names

#### Caution

Before you delete any prepackaged user files, be sure you have created at least one working user name with which to sign on. See Section 7, “Customizing User Environments,” for information about creating user names.



To maximize system security, you need to prevent users from signing on with generally known user names. To do so, delete or rename the following prepackaged user files that are included with applications:

- *[Sys]<Sys>.user* is the default user file supplied with Standard Software. It allows users to sign on by pressing GO, without requiring a user name or password.
- *[Sys]<Sys>CM.user* is a sample user file supplied with Context Manager.
- *[Sys]<Sys>Gps.user* is a sample user file supplied with the Generic Print System.

### Setting Up a Videoless Workstation Server

Workstations that use a protected-mode or virtual-memory server operating system can be set up as videoless servers. To set up a videoless workstation server, follow these major steps:

1. With the keyboard and monitor still connected to the workstation, make sure that system software is correctly installed and that the workstation functions normally as a server.
2. Edit *[Sys]<Sys>SysInit.jcl* so that RKVS and RUM are installed when the workstation is rebooted. (See “Cluster View,” in Section 4.)
3. Use the **Files** command to make sure that the file named *[Sys]<Sys>Vdm\_Ch.run* is present on the server. (If it is not, copy it from the VAM installation media.)
4. If you have an EISA server, use the System Configuration Utility (SCU) to disable the keyboard.
5. Remove the keyboard and monitor.

After the keyboard and monitor have been removed, you can access the server via Cluster View from a client workstation.

## Limiting Access to Commands

At some workplaces, you may want to limit access to certain commands to specified users only. For example, you may want to prevent anyone but yourself from changing the system date and time.

The Command Access Service allows you to limit access to certain commands that are executed on the server. The **Cluster View**, **Administrator Cluster View**, and **Set Time** commands are monitored by the Command Access Service.

The Command Access Service consists of a system service and a configuration file, which are described below.

### Installing the Command Access Service

You install the Command Access Service on the server. To install it during system initialization, add the following entry to your system initialization JCL file:

**Run [Sys]<Sys>AccessService.run, *LogFileSize*, *Password***

where

<i>LogFileSize</i>	Is the maximum number of sectors of disk space allowed for the log file. (See "Using the Command Access Service Log File," below.)
<i>Password</i>	Is a password that allows read access to the Command Access Service configuration file. (See "Protecting the Command Access Service Configuration File," below.)

On multiprocessor workstations and SRPs, you install the Command Access Service on the main processor only.

You can also install the Command Access Service with the **Install Command Access Service** command; see the *CTOS Executive Reference Manual*.



## Configuring the Command Access Service

The Command Access Service reads a configuration file named `[!Sys]<Sys>UserCmdsConfig.sys`. This file contains entries that monitor access to the recognized commands. A sample file is shown in Figure 6-1.

Note that users from other network nodes can be specified in the configuration file, as shown in Figure 6-1. See “Allowing Access to Users on Other Nodes,” below.

**Figure 6-1. Command Access Service Configuration File**

---

```

:SignInUserName:  Tricia
:AllowedCommands: 'Cluster View'
                  'Set Time'
                  'Administrator Cluster View'

:SignInUserName:  Jim
:AllowedCommands: 'Cluster View'

:SignInUserName:  {Accts}Renee
:AllowedCommands: 'Cluster View'

:SignInUserName:  Alex
    
```

---

## Creating the Command Access Service Configuration File

To create the configuration file, use the Editor application, as described in the following steps:

1. On the Executive command line, type **Editor**; then press **RETURN**.
2. Type `[!Sys]<Sys>UserCmdsConfig.sys`, as shown below.  
     [File name(s)]    [!Sys]<Sys>UserCmdsConfig.sys
3. Press **GO** twice.

4. Type the configuration file entries, as shown in Figure 6-1.

Valid values for *:AllowedCommands:* are 'Administrator Cluster View', 'Cluster View', and 'Set Time', as shown in Figure 6-1. Note that the 'Set Time' value applies to the *Date/Time* field in the SignOn form, as well as the **Set Time** command.

5. Press **FINISH**, then **GO**, to save the file and exit the Editor.

After you have created the Command Access Service configuration file, you can add user names or modify entries at any time, and the changes are effective immediately.

### Protecting the Command Access Service Configuration File

To prevent users from modifying the Command Access Service configuration file, assign it an access level of 1, 23, or 51 and a unique file password. See "Protecting Files," earlier in this section, for information about assigning passwords.

### Using the Command Access Service Log File

When a user executes a monitored command, the Command Access Service writes an entry to the log file, *[!Sys]<Sys>Logins.sys*. Each log-file entry contains the user's name, the date and time, and whether access was allowed, as shown in the following example:

```
Jim 8/13/92 10:30 AM: Cluster View - Access ALLOWED
```

If a user is not listed in *UserCmdsConfig.sys*, access is restricted, but nothing is logged when that user attempts to execute a monitored command. Therefore, to obtain information about users who attempt to execute monitored commands, you must include their user names in the configuration file. For example, in the sample configuration file, Alex's user name appears, but he is not allowed access to the monitored commands. Therefore, if he attempts to set the system date and time, the following entry is written to the log file:

```
Alex 8/13/92 10:30 AM: Set Time - Name only found
```

In the log file, new entries appear at the beginning of the file and old entries are dropped from the end of the file as it becomes full. To suppress logging, specify **0** as the log file size when you install the Command Access Service.

### Allowing Access to Users on Other Nodes

To allow access to a user on another node, specify the node name before the user name, as shown in the following example:

```
:SignOnUserName: {OtherNode}Ruth
```

Note that if a local user is also named Ruth, as shown in the above example, a separate entry without a node specification must be added for the local user. Only the **Cluster View** and **Administrator Cluster View** commands are monitored for access by remote nodes.



## Section 7

# Customizing User Environments

## What Is a User Configuration File?

A *user configuration file* (or simply a user file) defines a name to sign on with and a working environment, such as OFIS Document Designer, Context Manager™, or the Executive. It also contains information about the commands and applications that are available to the user.

You can create user files that suit your particular workplace, as demonstrated in the following examples:

- You can create user files for individual users using first names, last names, or nicknames. This method is often used when most users have their own workstations.
- You can create user names for the various tasks performed within your cluster, for example, writing or drawing. This method is sometimes used when workstations are shared among users.
- You can assign a user name to each workstation, for example, WS1, WS2, and so on. This is sometimes done to simplify signing on for large groups of users or when there are many diskless workstations.

## Creating a User File

To create a user configuration file, follow these steps:

1. On the Executive command line, type **User File Editor**; then press **RETURN**.

## Customizing User Environments

---

2. Enter a user name in the command form, as shown in the following example; then press **GO**.

```
User File Editor
User name      Freddie
[Old password]
[New password]
[Template file]
[Command file]
```

3. Press **GO** again to create the user name. The Functions menu appears:

Functions
<b>Modify Options</b>
Select Environment

603.7-A

4. On the Functions menu, position the highlight on *Modify Options*; then press **GO**.

A list of applications is displayed. This list may include applications that have not been installed on the workstation.

5. Position the highlight on the application you want to customize; then press **GO**.

A list of options and current values is displayed, for example, options for SignOn:

Option category: SignOn	
Volume	Sys
Directory	<b>Sys</b>
File prefix	
Password	
Node	
Text file	
User name	
Screen timeout	

603.7-B



6. Fill in the fields as you would an Executive command form. If you leave a field blank, a default value is used.
7. When you have completed the form, press **GO** to save your changes. If you do not want to save changes, press **CANCEL** to dismiss the form.
8. Repeat steps 5 through 7 to modify options for other applications.
9. Press **FINISH**, then **GO**, to save the file.

The user-file options for Standard Software are described later in this section. To learn about user-file options for other applications, see the appropriate manual. For more detailed information about the **User File Editor** command, see the *CTOS Executive Reference Manual*.

## Modifying a User File

To make changes to a user file, follow these steps:

1. On the Executive command line, type **User File Editor**; then press **RETURN**.
2. Enter the user name in the command form.
3. Press **GO**.

The Functions menu appears.

4. Proceed with step 5 under “Creating a User File,” above.

## Editing a User File Manually

To add an option that is not listed by the **User File Editor**, you use the Editor application. For example, you might need to add an option for a new software product.

The following subsections, “File Specifications for User Files” and “User File Format,” contain information you need to open and edit a user file with the Editor.

### File Specifications for User Files

To edit a user file with the Editor, you need to know its file specification. File specifications are written in the following format:

*[Sys]<Sys>Name.user*

where *Name* is the name the user signs on with, for example:

*[Sys]<Sys>Freddie.user*

User files are always stored in *[Sys]<Sys>*.

### User File Format

When you open a user file with the Editor, its appearance is similar to the example shown in Figure 7-1.

**Figure 7-1. User Configuration File**

---

```
:SignOnChainFile:[Sys]<Sys>Exec.run
:SignOnExitFile:[Sys]<Sys>Exec.run
:SignOnVolume:Sys
:SignOnDirectory:Sys
:SignOnFilePrefix:
:SignOnPassword:
:ExecCmdFile:[Sys]<Sys>Sys.cmds
```

---

Each line of a user configuration file is written in the following format:

*:Keyword:Value*

where

*:Keyword:* Is the name of a user file option; keywords and the placement of colons must not be changed. They correspond to the field names that are displayed when you select an application from the Options menu in the **User File Editor**.

*Value* Is the part you can change; values correspond to the entries you make in the **User File Editor**.

## Adding a User File Option

The following procedure describes how to use the Editor to add options to a user file. See "Editing a User File Manually," earlier in this section, for information about file specifications and formats for user files. See Section 4, "Using Administrative Tools," for more information about using the Editor.

1. On the Executive command line, type **Editor**; then press **RETURN**.
2. Type the file specification for the user file, as shown in the following example; then press **GO**.

```
[File name(s)]  [Sys]<Sys>Freddie.user
```

---

3. Position the cursor at the end of the file.
4. If necessary, press **RETURN** to move the cursor to a new line.
5. Type the keyword and value as shown in the following example. User file options are listed in the release documentation and manuals for applications.

```
:SignOnChainFile:[Sys]<Sys>Exec.run  
:SignOnExitFile:[Sys]<Sys>Exec.run  
:SignonVolume:d1  
:SignonDirectory:Work  
:SignonFilePrefix:  
:SignonPassword:  
:CookDinner:Yes
```

6. Press **FINISH**, then **GO**, to exit the Editor and save the file.

For changes to take effect, the user must log out and then sign on again.

### Creating a Working Environment

An environment consists of the application and other settings, such as path, that are started when the user signs on. It can also include initial settings that affect the mouse, electronic mail, and the Executive. The keywords that are recognized by the SignOn program and, therefore, can be included in an environment, are listed in Table 7-1.

Each user file contains a default environment, and optionally, one or more additional environments. To use an optional environment, the user enters its name in the *Environments* field of the SignOn form.

Figure 7-2 shows a sample user configuration file containing a default environment and an optional environment. The subsections that follow describe how to define environments in a user configuration file.

### Defining the Default Environment

The first set of entries in a user configuration file defines the default environment. In most cases, the default environment is set up to start the user's most frequently used application. To sign on to the default environment, the user leaves the *Environment* field of the SignOn form blank.

A default environment is shown at the beginning of the example user file in Figure 7-2. Note that the default environment does not need to begin with the *:Environment:* keyword. If it does, however, as in the case of some user files created with the **User File Editor**, the value for the *:Environment:* keyword is ignored by the SignOn program.

**Figure 7-2. User Configuration File: Multiple Environments**

---

A sample default environment is shown below.

```
:Environment:Default
:SignOnVolume:Mac0
:SignOnDirectory:Work
:SignOnPassword:Secret
:SignOnExitFile:[Sys]<Sys>Exec.Run
:SignOnChainFile:[Sys]<Sys>Exec.Run
:SignOnScreenTimeout:10
```

The following sample environment starts Context Manager.

```
:Environment:ContextManager
:SignOnVolume:Sys
:SignOnDirectory:Sys
:SignOnPassword:Secret
:SignOnExitFile:[Sys]<Sys>Exec.Run
:SignOnChainFile:[Sys]<Sys>CmInstall.Run
'Install Context Manager'
[Sys]<Sys>CmConfig.sys
:SignOnScreenTimeout:10
:MailUserName:Dooley MacMillan
:MailCenterName:PuliCenter
:MailNotification:Normal
```

The following entries are not recognized by SignOn; they should be specified one time only, at the end of the user configuration file.

```
:EditorOpenMode:Read
:EditorTabLength:4
:EditorLineMapping:LF
:GpsUserClass:Admin
:ClusterViewPartitionSizeInK:1280
```

---



### Defining Optional Environments

A user configuration file may contain one or more optional environments. An optional environment allows the user to sign on with his or her usual user name, but to start an application that is different from the default environment. This is particularly helpful when using an application such as OFIS Document Designer, which stores macros and phrases on a user name basis, or GPS Print Manager, which assigns administrative privileges based on a user file entry.

The *:Environment:* keyword introduces each optional environment, as shown near the middle of the example file in Figure 7-2. To create an optional environment, add its keywords and values with the Editor application, as described earlier in this section.

In optional environments, use only those keywords that are recognized by the SignOn program, as listed in Table 7-1. Place keywords that are read by other applications at the end of the user configuration file; each should appear one time only, as shown near the end of the example file in Figure 7-2. Such keywords include those that affect the Editor, Cluster View, GPS, and applications such as OFIS Document Designer.

To sign on to an optional environment, the user specifies the value for the *:Environment:* keyword in the *Environment* field of the SignOn form.

#### Caution

Do not use the **User File Editor** command to modify a user file that contains optional environments. Doing so deletes the keywords and values for optional environments from the user configuration file. Instead, edit such user files manually, as described earlier in this section.

### Keywords Recognized by the SignOn Program

Table 7-1 lists the user-file keywords that are recognized by the SignOn program. See "User File Options for Standard Software," later in this section, for a detailed description of keywords for the Standard Software utilities. For a description of keywords that affect electronic mail, see the documentation for your mail application.



When Context Manager is defined as the environment, entries in the Context Manager configuration file may supersede some of the SignOn keyword values in the user configuration file.

The *:SignOnChainFile:* and *:SignOnExitFile:* keywords determine the applications that are started when the user signs on and exits. They are described in more detail later in this section.

**Table 7-1. User File Keywords Recognized by SignOn**

<b>Keyword</b>	<b>Default Value</b>
<i>:Environment:</i>	Default environment
<i>:ExecCmdFile:</i>	<i>[Sys]&lt;Sys&gt;Sys.cmds</i>
<i>:MailCenterName:</i>	None
<i>:MailNotification:</i>	None
<i>:MailPassword:</i>	None
<i>:MailUserName:</i>	None
<i>:MouseAcceleration:</i>	No change
<i>:MouseSpeed:</i>	No change
<i>:SignOnChainFile:</i>	<i>[Sys]&lt;Sys&gt;Exec.run</i>
<i>:SignOnDirectory:</i>	<i>Sys</i>
<i>:SignOnExitFile:</i>	<i>[Sys]&lt;Sys&gt;Exec.run</i>
<i>:SignOnFilePrefix:</i>	None
<i>:SignOnNode:</i>	None
<i>:SignOnPassword:</i>	No change
<i>:SignOnTextFile:</i>	None
<i>:SignOnScreenTimeout:</i>	Always on
<i>:SignOnUserName:</i>	No change
<i>:SignOnVolume:</i>	<i>Sys</i>

### The SignOn Chain File Entry

The *:SignOnChainFile:* keyword defines the application that is started when the user signs on. The following example shows an entry that starts OFIS Document Designer:

```
:SignOnChainFile: [Sys]<Sys>OFISDocumentDesigner.run
```

The following example shows a user file entry for a Context Manager environment. To specify parameter values for the chain file, you type the Executive command name and then the parameter values as you would enter them in an Executive command form.

```
:SignOnChainFile: [Sys]<Sys>CmInstall.run  
'Install Context Manager'  
[Sys]<Sys>CustomCmConfig.sys
```

### The SignOn Exit File Entry

The *:SignOnExitFile:* keyword defines the program that appears when the user exits an application. The following entry exits to the Executive:

```
:SignOnExitFile: [Sys]<Sys>Exec.run
```

To exit to the SignOn screen, specify *[Sys]<Sys>SignOn.run*.

The SignOn exit file can limit the user to a single application. For example, if you set up an OFIS Designer environment that exits to the SignOn screen, the user does not have access to the Executive or other applications on the system.

## Limiting Access to the System

When you install Standard Software, a default user file, named *[Sys]<Sys>.user*, is created. It allows users to sign on by pressing GO, without entering a user name in the SignOn form. You can effectively limit access to the system by removing the default user file and then assigning passwords to all other user files, as described in the following sections.

## Removing the Default User File

### Caution

Make sure that you have created a user name you can sign on with before you remove the default user file. See "Creating a User File," earlier in this section.

To remove the default user file, use the **Delete** command, as shown below:

```
Delete
File list          [Sys]<Sys>.user
[Confirm each?]   _____
```

## Assigning Passwords to User Names

If your system is password protected, you can prevent users from signing on with each other's user names. To do this in user files that do not contain multiple environments, use the **User File Editor** command to assign a password to each user file, as described below.

1. Start the **User File Editor** command, as described earlier in this section.
2. Fill in the command form, as shown in the following example:

```
User File Editor
User name          Freddie
[Old password]    _____
[New password]    mypswd
[Template file]   _____
[Command file]   _____
```

3. Press **GO**.
4. Select *Modify Options*; then press **GO**.
5. Select *SignOn*; then press **GO**.

6. Enter the volume or directory password in the *SignOn Password* field.
7. Press **FINISH**; then press **GO** to save the user file and exit the **User File Editor**.

The password you assign with the **User File Editor** command is entered by the user in the SignOn form when the user signs on. It is a file-level password with protection level 51. See Section 6, "Implementing System Security," for information about password protection levels.

Alternatively, in user files that contain multiple environments, use the **Set Protection** command to assign a password and protection level to the user file. Then, use the Editor to specify the password you just assigned as the value for the *:SignOnPassword:* parameter in the user configuration file.

## Signing On Automatically

You can set up a workstation so that it signs on automatically with a specified user name, password, and environment when it is rebooted. To use this feature, make the following entry in the system initialization JCL file:

```
Run [Sys]<Sys>SignonBE.run,UserName,UserFile,Pswd,Envir
```

where

<i>UserName</i>	Is the user's SignOn user name. The default is no user name.
<i>UserFile</i>	Is the file specification of the user configuration file to be read, if that is different from <i>[Sys]&lt;Sys&gt;UserName.user</i> . (See the description of the <i>:SignOnUserName:</i> entry, later in this section.) The default file is <i>[Sys]&lt;Sys&gt;UserName.user</i> .
<i>Pswd</i>	Is a valid password for the user configuration file. The default is no password.
<i>Envir</i>	Is the name of the environment to be started. The default is the first environment in the specified user configuration file. (See "Creating a Working Environment," earlier in this section.)

The automatic SignOn entry is placed directly before the *End* statement, as shown in the following example:

```
Job SysInit
Command Install Mouse Service
Run [Sys]<Sys>SignonBE.run,Wink,,mypswd,OFDD
End
```

See Section 9, "Installing System Services," for detailed information about system initialization JCL files for workstations and SRPs.

## Signing On With a Magnetic Card Reader

You can set up a workstation so that users sign on by inserting a card into a magnetic card reader. For a workstation to recognize a magnetic card reader, the MCR Service must be installed during system initialization. To do that, add the following entry to the system initialization JCL file:

### **Command Install MCR Service**

See Section 9, "Installing System Services," for more detailed information about the system initialization process.

The magnetic card used for signing on must contain exactly those keystrokes that would be entered in the SignOn form, including cursor movement and execution keys.

It is not advisable to include date and time information on a magnetic card. If the date and time need to be set routinely, omit the **GO** keystroke from the card. The user then enters the date and time in the SignOn form and presses **GO** manually to complete the procedure.



# User File Options for Standard Software

The following Standard Software commands and applications include user file options:

- SignOn program
- Executive
- Mouse Service
- **Installation Manager** command
- **Cluster View** and **Administrator Cluster View** commands
- Editor application

The options for these programs are described below. Both the field names displayed by the **User File Editor** command and the literal keywords are listed.

## SignOn Options

The SignOn options take effect when a user signs on. They remain in effect until changed by the user or until superseded by the Context Manager configuration file (see your Context Manager manual).

### ***Volume***

Keyword: *:SignOnVolume:*

Default: *Sys*

This entry defines a volume for the default path setting. Specify a volume or device name.

### ***Directory***

Keyword: *:SignOnDirectory:*

Default: *Sys*

This entry defines a directory for the default path setting. Specify a directory name.



### ***File prefix***

Keyword: *:SignOnFilePrefix:*

Default: None

This entry defines a file prefix for the default path setting. Specify the characters you want to use as a file prefix.

### ***Password***

Keyword: *:SignOnPassword:*

Default: See below

This entry defines the password that takes effect after the user signs on. Specify a volume or directory password.

If you specify a password, it supersedes the password entered in the SignOn form. If you do not specify a password, the password entered in the SignOn form becomes the default password. (See “Limiting Access to the System,” earlier in this section, and Section 6, “Implementing System Security.”)

### ***User name***

Keyword: *:SignOnUserName:*

Default: User name entered in the SignOn form

This entry defines the user name that takes effect after the user signs on. A user name specified here supersedes the user name entered in the SignOn form. It then appears as the active user name in the status area of the Executive screen, and user file options are read from the user configuration file that corresponds to it. If such a user configuration file does not exist, unpredictable behavior or errors may occur.

### *Text file*

Keyword: *:SignOnTextFile:*

Default: None

This entry defines a file, the contents of which are displayed on the screen when the user signs on. Enter the full file specification for the file you want to display.

### *Screen time out*

Keyword: *:SignOnScreenTimeout:*

Default: Always on

This entry defines the elapsed number of minutes before the screen is to be shut off when a workstation is not being used. Enter a number from 1 to 100.

## **Executive Option**

The following option applies to the Executive application.

### *Command file*

Keyword: *:ExecCmdFile:*

Default: *[Sys]<Sys>Sys.cmds*

This entry defines the command file to be used by the Executive. Enter the full file specification for a valid Executive command file.

## **Mouse Options**

The following options apply to the mouse and are implemented only when the Mouse Service is installed before you sign on. To display these options with the **User File Editor** command, select General from the Modify Options menu.

**Note:** *More mouse options are available for certain applications, such as Art Designer. See the documentation for each application for more information about user file options.*

### ***Left-handed user?***

Keyword: *:LeftHanded:*

Default: No

This entry reverses the functions of the mouse buttons for a left-handed user. Specify **Yes** to reverse the mouse button functions.

### ***Setting for speed of the mouse***

Keyword: *:MouseSpeed:*

Default: 4

This entry defines the speed at which the mouse cursor moves as you move the mouse. Enter a number between 1 (slowest) and 10 (fastest).

## **Installation Manager Options**

The following options define initial values for the **Installation Manager** command. You can change them later during the software installation procedure. See “Using the Installation Manager,” in Section 8, “Installing Applications,” for more information.

### ***Install public?***

Keyword: *:InstallPublic:*

Default: No

This entry defines whether software is installed on the server (public) or on the local workstation only (private). Enter **Yes** for public installations; enter **No** for private installations.

### ***Install verbose?***

Keyword: *:InstallVerbose:*

Default: No

This entry defines the type of messages displayed during software installation. Enter **Yes** to display the entire installation script (verbose). Enter **No** to display only selected status messages (silent).

### ***Install backup?***

Keyword: *:InstallBackup:*

Default: No

This entry specifies whether the current version of an application is backed up before new software is installed. Enter **Yes** to back up a current version. Enter **No** to bypass the backup operation.

### ***Save backup?***

Keyword: *:SaveBackup:*

Default: No

This entry specifies whether the backup (see *Install Backup?*, above) is saved after installation has been completed.

### ***Use log file?***

Keyword: *:InstallLogFile:*

Default: No

This entry specifies whether an installation log file is created.

### ***Log file name***

Keyword: *:InstallLogFileName:*

Default: *[Sys]<Installed>Install.log*

This entry defines a log file for command output. Enter a full file specification for the log file, including volume, directory, and file name.

### ***Archive path (private)***

Keyword: *:InstallArchivePath:*

Default: *[Sys]<Installed>*

This entry defines the volume and directory where the backup will be created during private installations. Enter a volume and directory name, including brackets, for example, *[Sys]<Installed>*.

### ***Archive path (public)***

Keyword: *:InstallPublicArchivePath:*

Default: *[!Sys]<Installed>*

This entry defines the volume and directory where the backup will be created during public installations. Enter a volume and directory name, including brackets, for example, *[!Sys]<Installed>*.

### ***Destination volume (private)***

Keyword: *:InstallVolume:*

Default: *[Sys]*

This entry defines the volume where software will be installed during private installations.

### ***Destination volume (public)***

Keyword: *:InstallPublicVolume:*

Default: *[!Sys]*

This entry defines the volume where software will be installed during public installations.

### ***CM Config File (Private)***

Keyword: *:InstallCmFile:*

Default: *[Sys]<Sys>CmConfig.sys*

This entry defines the Context Manager configuration file that will be updated during private installations. Enter a full file specification, including volume, directory, and file name.

### ***CM config file (public)***

Keyword: *:InstallCmFilePublic:*

Default: *[!Sys]<Sys>CmConfig.sys*

This entry defines the Context Manager configuration file that will be updated during public installations. Enter a full file specification, including volume, directory, and file name.



### Cluster View Options

The following options define default values for the Cluster View commands. You can override defaults in the Executive command form when you execute the **Cluster View** or **Administrator Cluster View** command. See “Cluster View,” in Section 4, “Using Administrative Tools,” for more information.

#### *Processor name*

Keyword: *:ClusterViewProcessorName:*

Default: See below

This entry defines the SRP processor on which the Cluster View session is started. Enter the four-character processor ID, for example, GP00.

On SRPs and multiprocessor workstations, the default is the processor to which the workstation is connected.

#### *Run file*

Keyword: *:ClusterViewRunFile:*

Default: *[Sys]<Sys>Exec.run*

This field applies to processors running the Remote User Manager. Its entry defines a run file to be started on the specified processor. Enter the file specification of the run file.

#### *Partition size in Kbytes*

Keyword: *:ClusterViewPartitionSizeInK:*

Default: 400

This field applies to processors running the Remote User Manager. Its entry defines a partition size for the run file named above. Enter the number of K bytes for the partition.



***Default node***

Keyword: *:ClusterViewPathNode:*

Default: Server

This entry defines the network node on which Cluster View will be executed. Enter the name of a network node.

***Default volume***

Keyword: *:ClusterViewPathVolume:*

Default: Sys

This entry defines a default volume for the Remote User Manager sessions. Enter the name of a volume on the server or network node.

***Default directory***

Keyword: *:ClusterViewPathDirectory:*

Default: Sys

This entry defines a default directory for Remote User Manager sessions. Enter the name of a directory on the server or network node.

***Default file prefix***

Keyword: *:ClusterViewFilePrefix:*

Default: None

This entry defines a default file prefix for Remote User Manager sessions. Enter the characters you want to use for a default file prefix.

***Default password***

Keyword: *:ClusterViewPathPassword:*

Default: Default password

This entry defines a default password for Remote User Manager sessions. Enter a valid password for the server or network node.

### Editor Option

The following option applies to the Editor. See the *CTOS Editor User's Guide* for more information.

#### *Display width*

Keyword: *:Editor80Column:*

Default: 79 columns

This entry defines the width of the Editor screen display. By default the screen shows 79 columns. You can also set the Editor to display 80 columns during any Editor session by pressing **F7** to bring up the Set Defaults menu and selecting 80 column mode.

# Section 8

## Installing Applications

### Software Packages

Applications, such as word processing programs and spreadsheets, are distributed in software packages that consist of the following:

Distribution media	Distribution media are floppy disk, QIC tapes, and compact discs (CDs) that contain the programs, commands, and configuration files for an application.
Release documentation	Release documentation accompanies the distribution media and contains the most current information about the product. It is called a Software Release Announcement (SRA), Release Notes, or Release Notice.
Manuals	A manual or set of manuals is available for most software products. Manuals contain detailed information about configuring and using applications.

### Planning the Installation

By planning the installation of software applications, you speed up the process and help to ensure that you install everything that is needed on the system. This is particularly true when you are setting up a new system.

Before you begin, decide which applications will be needed on your cluster; some of the most frequently used applications are described later in this section. Then, determine where in the cluster you need to install the software, for example, on the server only, or on all client workstations, as well.

## Installing Applications

---

Check the release documentation to be sure that the workstation or SRP meets the requirements for the product. For example, each application requires a certain amount of disk space and memory. Make sure that enough of those resources are available before you begin installing software.

## What Applications Are Available?

Some frequently used applications are briefly described below. In addition, programming languages, such as Pascal, COBOL, and C, as well as a large number of applications from other software firms, are available for CTOS workstations. See your sales representative for detailed information about software products.

### Office Automation Applications

OFIS Graphics	For creating charts and drawings
OFIS Mail	For sending and receiving electronic mail messages
OFIS Document Designer	For word processing and office publishing
OFIS Imager	For scanning pictures and editing and printing scanned images
OFIS Spreadsheet	For accounting and financial planning

### Communications Applications

BNet	For connecting clusters together to form a network
Cluster Access	For accessing files and other resources on client workstations in a cluster
LANce	For adding PC-compatible systems to a CTOS cluster

### Other Applications

Context Manager	For starting and running multiple applications on a workstation
Generic Print System (GPS)	For providing access to a wide variety of printing devices, including laser printers, scanners, and plotters

### Using the Installation Manager

You install software applications with the **Installation Manager** command, which is packaged with Standard Software. It provides the following features for software installation:

- Installation from floppy disks
- Installation from QIC tape and compact disc (certain products only)
- Installation from the server to client workstations
- Public installations onto the server for use by client workstations
- A log file for tracking installation errors
- Removal of software packages
- Recovery from installation failure

These features are described in more detail later in this section.

Some applications were released before the **Installation Manager** command existed and do not take advantage its features. As new versions of applications are released, however, installation procedures are updated. Always check the release documentation or software installation guide for the most current installation instructions.

# Installing Applications

*Note:* The installation instructions for some older applications tell you to use the **Install**, **Software Installation**, or **XESoftware Installation** commands. Those commands are now obsolete, and their functionality has been replaced by the **Installation Manager** command. **Installation Manager** features, such as public installation and installing from the server, however, are not available to those applications.

The following procedure describes how to install applications with the **Installation Manager** command. Before you begin, make sure that Context Manager is not running on the workstations and, if you are installing on the server, that the application you are updating is not currently in use by other workstations in the cluster.

1. On the Executive command line, type **Installation Manager**; then press **RETURN** to display the command form:

```
Installation Manager
[Non-interactive?] _____
[Append to log file?] _____
```

2. Fill in the command form according to your needs. (In most cases, you can leave it blank.) Parameter fields are described in Table 8-1.
3. Press **GO**.

The Software Operation menu appears:

**Software Operation**

Press GO to select a menu item, CANCEL or FINISH to quit

- Install new software**
- Show installed software
- Remove installed software

603.8-A



4. Position the highlight on *Install new software*; then press **GO**.

The Install Media menu appears:

**Install Media**

Press GO to select an item, CANCEL for main menu, FINISH to quit

**Floppy Installation**

Tape installation

Install from server

CD-ROM Installation

603.8-B

5. Position the highlight on the distribution media you want to select; then press **GO**. (The *Install from server* option does not appear on standalone or server workstations.)

The Installation Defaults menu appears, as shown below. From this menu, you can continue the installation with default parameters, or you can examine and change parameters. Both options are described below.

**Installation Defaults**

Press GO to select item, CANCEL for main menu, FINISH to quit

**Continue Installation**

Examine/Change Defaults

603.8-C

- To continue the installation using default parameters, position the highlight on *Continue Installation*; then press **GO**.

## Installing Applications

---

- To examine or change default values, position the highlight on *Examine / Change Defaults*; then press **GO**. The Installation Parameters form appears, as shown below. Default values, which come from the user configuration file, are highlighted.

Installation Parameters		
Fill in or modify, press GO to accept, CANCEL to dismiss		
Public	Yes	No
Verbose	Yes	No
Backup previous version	Yes	No
Save copy of backup	Yes	No
Save defaults in user file	Yes	No
Press Y or N, or use arrow keys		

603.8-D

- a. To change a default parameter, position the highlight on the line you want to change.
  - b. Select Yes or No by typing **Y** or **N** or moving the **RIGHT ARROW** and **LEFT ARROW** keys. Software installation parameters are described in Table 8-2.
6. Press **GO** to continue the installation.  
Depending on the type of installation you choose, a number of interactive fields may be displayed.
  7. Fill in fields as they are displayed, pressing **GO** after each. Interactive fields are described in Table 8-3.  
The installation begins, and you are informed when it is complete.
  8. Reboot the workstation.

Table 8-1. Installation Manager Command-Form Parameters

---

Field Name	Description
<i>[Non-interactive?]</i>	Default: No  Enter <b>Yes</b> if you do not want the installation script to pause for user input. If you enter <b>No</b> or leave this field blank, the installation script may pause so that you can select options or enter input.  <b>Note:</b> <i>This field does not pertain to floppy disk installation procedures.</i>
<i>[Append to log file?]</i>	Default: No  Enter <b>Yes</b> if you want to append to, rather than overwrite, an existing log file. If you enter <b>No</b> or leave this field blank, the installation log file is overwritten with each successive installation procedure.

---

**Table 8-2. Installation Manager Parameters-Form Selections**

Parameter	Description
<i>Public</i>	<p>Select <b>No</b> for a <i>private installation</i>, which installs software on the local workstation.</p> <p>Select <b>Yes</b> for a <i>public installation</i>, which installs software on the server. Publicly installed software can be used by all workstations in the cluster.</p>
<i>Verbose</i>	<p>Select <b>No</b> for a <i>silent installation</i>. During a silent installation, only a few progress messages are displayed, but all command output is written to the installation log file.</p> <p>Select <b>Yes</b> for a <i>verbose installation</i>. During a verbose installation, all messages and command output are displayed and some of them are written to the installation log file.</p>
<i>Backup previous version</i>	<p>Select <b>No</b> if you do not want to back up the current version of the application you are installing.</p> <p>Select <b>Yes</b> to back up the current version of a software product before the new version is installed. When you do this, the backed up version can be restored if the installation fails for any reason.</p>
<i>Save copy of backup</i>	<p>Select <b>No</b> if you do not want to save a copy of the backup.</p> <p>Select <b>Yes</b> if you want to save a copy of the backup. You can use this backup later to restore the previous version if necessary, for example, if you discover that the newer version isn't compatible with your hardware or software configuration.</p>
<i>Save defaults in user file</i>	<p>Select <b>Yes</b> to change your user file entries to the parameters you just selected. They become new defaults for the <b>Installation Manager</b> command.</p> <p>Select <b>No</b> if you do not want to change your user file.</p>

Table 8-3. Installation Manager Interactive Fields

Field Name	Description
<i>Archive path</i>	This field appears if you specified <b>Yes</b> to <i>Backup previous version?</i> . It displays the volume and directory to which the backup is to be written. Change it if you want to back up to a different volume and directory.
<i>Software destination</i>	Default: <i>[Sys]</i> (private) or <i>[!Sys]</i> (public)  This field shows the volume where the application is to be installed. Change it if you want to install the application on a different volume.
<i>Password</i>	This field is displayed if a password is required to access the software destination directory. The installation will not continue until a valid password is supplied.
<i>Tape spec</i>	This field appears if you are performing a tape installation. Enter the tape specification for the application you want to install. It consists of the name of the tape drive in which the distribution media is inserted and, optionally, a tape mark number. See "Specifying a Tape Drive," in the <i>CTOS Executive Reference Manual</i> , for information about tape specifications.
<i>User name</i>	This field shows the user name with which you are currently signed on; during software installation, this is frequently not your usual user name. If you specified <b>Yes</b> to <i>Save defaults in user file</i> , enter the user name for which you want to save default values.
<i>CmConfig file</i>	If Context Manager is installed during the installation, the name of the active CM configuration file appears in this field. If Context Manager is not installed, the file name <i>[Sys]&lt;Sys&gt;CmConfig.sys</i> is displayed. If the displayed file name is not your usual CM configuration file, enter the appropriate file specification.  <b>Note:</b> <i>Whether an application is added to Context Manager during the installation procedure is dependent on the installation script for the application. This field simply identifies the appropriate CM configuration file.</i>
<i>Command file</i>	This field appears if you are performing a private installation. It displays the file name of the active Executive command file. If you want new commands placed in a different command file, enter its file specification.

# Installation Manager Features

The **Installation Manager** includes many features, which are only briefly described on the preceding pages. The following sections contain more detailed information about tailoring software installation to the needs of your cluster.

## Installing From Floppy Disks

Most software applications are distributed on floppy disks. To install an application from floppy disks, select *Floppy installation* from the Install Media menu.

## Installing From QIC Tape

Some applications are distributed on QIC tape as well as on floppy disk. Applications on QIC tape can be installed from a tape drive on either the local workstation or the server. To install an application from QIC tape, select *Tape installation* from the Install Media menu.

## Installing From Compact Disc

An entire set of operating systems and application software is distributed on a single compact disc (CD). If the server is equipped with a CD-ROM drive and is running CTOS III software, you can install from the CD onto workstations throughout the cluster. To install an application from CD, select *CD-ROM installation* from the Install Media menu.

## Installing From the Server

After you have installed an application for public use on the server, it can be installed on client workstations over cluster lines. This has the following advantages over other installation media:

- It is convenient; users do not have to locate the distribution media.
- It installs only what is necessary for client workstations.
- The system administrator can configure the application appropriately before other users in the cluster install it.



To install an application from the server, select *Install from server* from the Install Media menu. See also “Installing Public Software,” below.

### Installing Public Software

Public software is installed on the server for use by all workstations in the cluster. You can perform a public installation from a client workstation or on the server. When you perform a public installation, application programs are copied to the server and commands are created in a command file on the server named *[!Sys]<Sys>Cluster.cmds*.

When a user issues a command, the Executive reads the public command file if the command name is not present in the user’s private command file. Therefore, all client workstations automatically have access to publicly installed software; no preparation or setup is required. In addition, software packages must be installed publicly to make them available to client workstations for installation from the server.

To install public software, specify **Yes** to *Public* on the Installation Parameters menu.

### Using the Log File

In most cases, a log file named *[Sys]<Installed>Install.log* is automatically created during an installation procedure. It contains a record of command output for the installation. If an installation fails, the log file may contain error messages that can help you determine the cause of the failure.

To conserve disk space, the log file is overwritten with each installation procedure. You can, however, choose to append to the existing log file, by specifying **Yes** in the *[Append to log file?]* field, when you start the **Installation Manager** command. This can be helpful when you are troubleshooting installation problems.

The log file can also be disabled or its name changed by making an entry in the user configuration file. For more information, see “Installation Manager Options,” in Section 7, “Customizing User Environments.”

### Removing an Application

With the **Installation Manager** command, you can remove an application from a system. That includes the programs, configuration files, and commands that are associated with the application.

#### Caution

Do not remove the Basic System Utilities, the CTOS operating system, or the video package (VAM). They are required to boot the system.

To remove an application, start the **Installation Manager**; then follow these steps:

1. From the Software Operation menu, position the highlight on *Remove installed software*; then press **GO**.
2. Select *Public* or *Private* to display the appropriate list of software to deinstall.
3. Position the highlight on the application you want to remove; then press **GO**.

### Recovering From Installation Failures

When selecting installation parameters, you can choose to back up the currently installed application. The backup takes place before any new software is installed, and it can be restored if the installation fails for any reason.

To choose this option, specify **Yes to Backup previous version** on the Installation Parameters menu.

To restore the older version if the installation fails, press **GO** when the following message is displayed:

Press **GO** to restore the backup, **CANCEL** to finish.

To determine the cause of the failure, see "Common Installation Problems," later in this section, and the release documentation for the application.

## Restarting an Installation

If you do not restore the previous version after an installation failure, you might be able to restart the installation procedure at the point from which it failed. This eliminates the tedium of repeating an entire installation procedure.

***Note:** The restart feature is not available for all applications. In addition, the restart feature is not available after certain nonrecoverable errors occur.*

The following procedure describes how to restart an installation. Before you restart it, though, correct the problem that previously caused the installation to fail.

1. Start the **Installation Manager** command, as described earlier in this section.

The following message is displayed:

```
The previous installation of Product Name failed.
```

```
Press GO to restart installation, CANCEL to start new installation.
```

2. Do one of the following:
  - Press **GO** to restart the installation from the point at which it failed.
  - Press **CANCEL** to start the installation again from the beginning.

## Loadable Requests

*Loadable requests* expand the capabilities of the operating system and may be required by certain system services and applications. When required, loadable request sets are packaged on the installation media for the software that requires them.

## Installing Applications

---

During software installation, loadable requests are merged into a file named *[Sys]<Sys>Request.sys*, which is loaded into memory when a workstation or SRP boots. If you bypass part of the installation procedure, however, the loadable requests may not be merged. This can result in Error 31 (no such request) when you try to install a system service or use an application.

On SRPs, you can specify a request file other than *[Sys]<Sys>Request.sys* in the operating system configuration file. If you do this, though, you may need to merge requests manually into the file you specify. See the **List Request Set**, **Make Request Set**, and **Update Request Set** commands in the *CTOS Executive Reference Manual*.

See the *:LoadableRequestFile:* parameter, in Section 16, "Configuring Operating Systems," for information about specifying a request file other than *[Sys]<Sys>Request.sys*.

## Common Installation Problems

Table 8-4 describes some common problems that can occur during software installation. See also the Software Release Announcement for the product you are installing.

Table 8-4. Software Installation Errors

Error Code	Description
202	<p>Directory full</p> <p>This error occurs if a directory becomes full during software installation. Check the directory named <i>&lt;installed&gt;</i> (on either the private or public volume), as well as the directory into which you are installing software.</p>
219	<p>Access denied</p> <p>This error occurs if the destination directory requires a different password from <i>[Sys]&lt;Sys&gt;</i>. For example, if you must specify <b>PswdA</b> to access the software destination volume, but <b>PswdB</b> is required to access <i>[Sys]&lt;Sys&gt;</i>, the <b>Installation Manager</b> may not be able to create new commands in <i>[Sys]&lt;Sys&gt;</i>. Similarly, if the destination volume does not require a password, you will not be prompted to enter one; however, if a password is required for <i>[Sys]&lt;Sys&gt;</i>, the installation may fail.</p> <p>To avoid such problems, use the <b>Set Directory Protection</b> command to remove the directory password from <i>[Sys]&lt;Sys&gt;</i> before you begin the installation. Then, after the installation is complete, use the <b>Set Directory Protection</b> command again to reassign the original password to <i>[Sys]&lt;Sys&gt;</i>.</p>
220	<p>File in use</p> <p>This error occurs when the Executive command file for a workstation is in use. This can happen if the workstation is running Context Manager or if several workstations are sharing a command file on the server. If the workstation is running Context Manager, log out and after you sign on again, restart and perform the installation before you install Context Manager. For a shared command file, have all users who share the command file log out and remain logged out while you restart and perform the installation.</p>
230	<p>Disk full</p> <p>This error occurs when you do not have enough disk space for the application. You will have to make room on the disk before you restart the installation.</p>





## Section 9

# Installing System Services

## What Is a System Service?

A *system service* is a software program that manages or provides access to a resource that is available on an individual workstation or within the cluster. A resource is frequently a hardware device, such as a mouse, a printer, or a tape drive; or it can be a software component, such as a communications gateway, a database, or an electronic mail center.

Some system services, such as those that manage the keyboard and file system, are part of the operating system, so you do not need to install them or be aware of them as system services. Because system services consume memory, however, only those that are essential are included within the operating system itself. You must install additional system services to handle optional resources, such as printers and modems.

Installing system services is a separate task from installing software applications. When you install applications, you copy programs and commands onto a disk. When you install a system service, you load a program from a disk into memory on a processor. The programs and commands you use to install system services are placed on your disk when you install software applications. Many applications require system services.

As a system administrator, you determine where in the cluster system services are required, and you set up the installation procedures. This section describes some commonly used system services and includes procedures for installing them on workstations and SRPs.

# What System Services Do You Need?

The system services you need to install depend on the equipment and applications being used within your cluster. Some frequently used system services are described below. A server may require almost all of them, while a client workstation may not need any. More information about where to install system services is included later in this section.

## Standard Software System Services

The following system services are packaged with Standard Software.

CD-ROM Service	Required for a CD-ROM drive.
Cluster File Access Services	Allow users to access files on client workstations, as well as on the server.
Command Access Service	Limits access to the <b>Administrator Cluster View, Cluster View, and Set Time</b> commands to specified users only.
DataComm Service	Required for a DCX port expander module.
Math Service	Performs floating-point calculations for applications that do not include that function.
MCR Service	Required for a magnetic card reader.
Mouse Service	Required for a mouse.
Queue Manager	Required for spooled printing. It is also used to control message passing by other applications, such as Batch and SNA RJE. The Queue Manager is always installed on the server.
Remote Keyboard Video Service	Required for Cluster View. On SRPs and IOP cards, it is automatically loaded when the system boots. On other workstation processors, it is installed as a system service.

Remote User Manager (RUM)	Allows multiple Cluster View sessions to be started on a protected-mode processor.
Screen Print Service	Required to print the contents of the screen to a file or a printer.
Sequential Access Service	Required for digital data storage (DDS) drives, quarter-inch cartridge (QIC) tape drives, and half-inch tape drives.
Spooler	Required for pre-GPS spooled printing services. (See the appendix about pre-GPS printing in the <i>CTOS Generic Print System Administration Guide</i> .)
Statistics Service	Provides information about memory utilization and disk space.
Voice Service	Required to send or receive voice data messages.
XBIF Service	Required for certain workstation modules to communicate across the X-Bus. Other module-specific system services, such as the XC-002 Service, must be installed after the XBIF Service.
XC-002 Service	Required for an XC-002 Port Expander module.

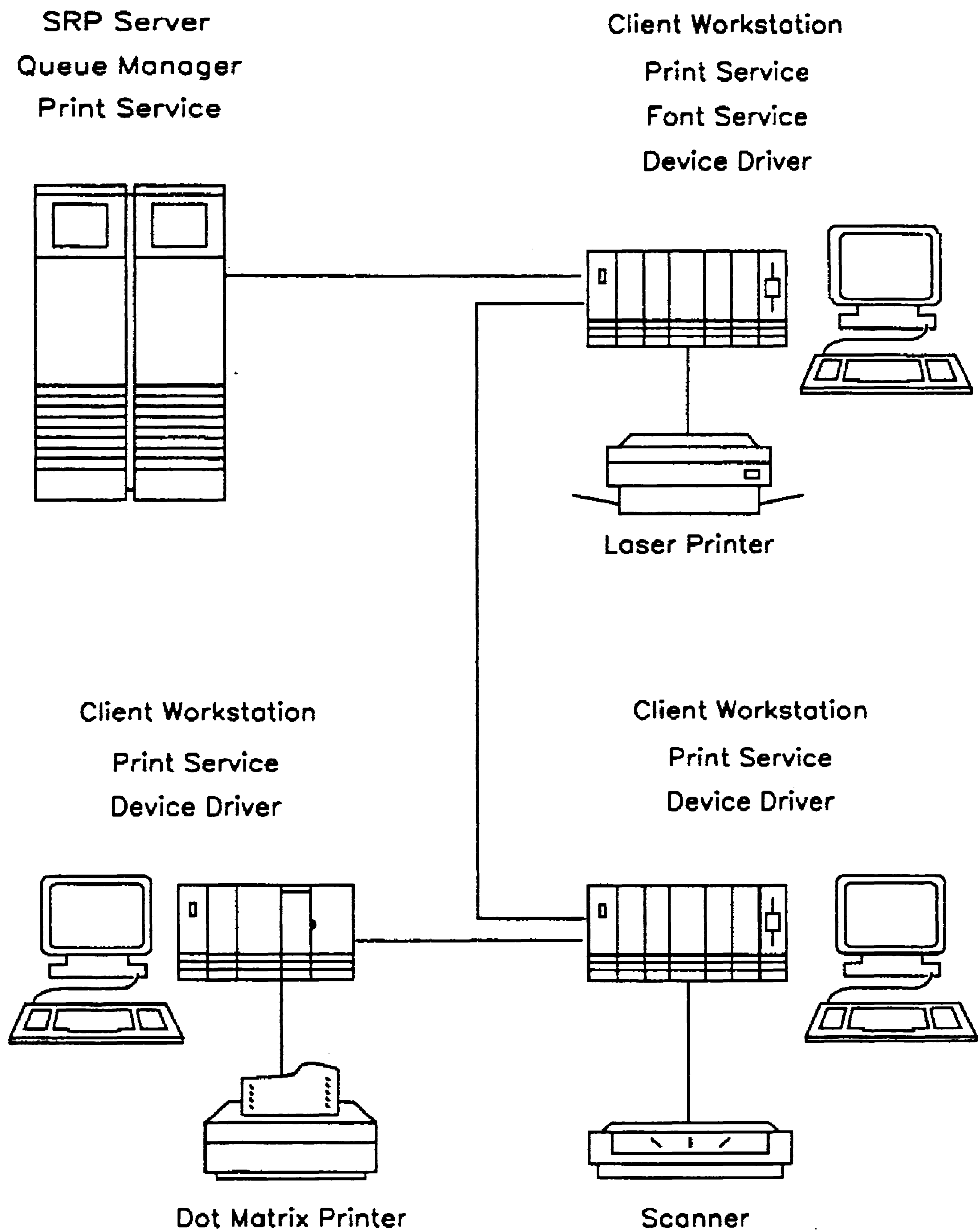
### Generic Print System Services

The Generic Print System (GPS), although packaged separately from Standard Software, is usually installed and maintained by the system administrator. It includes the following system services:

Queue Manager	This is the same system service supplied with Standard Software. It is installed on the server, and only one Queue Manager per cluster is installed.
Print Service	This system service is installed on the server and on each workstation or processor to which a spooled printer is attached.
Font Service	This system service can be installed on the server for use by client workstations or on each client workstation that is using a font database. Performance is improved when the Font Service is installed on the client workstation.
Printer device drivers	Each of these system services controls a specific type of printer and is installed on the processor to which the printer is attached.

Figure 9-1 shows an example of GPS system services installed within a cluster. See the *CTOS Generic Print System Administration Guide* for more detailed information.

Figure 9-1. GPS System Services in a Cluster



### Electronic Mail Services

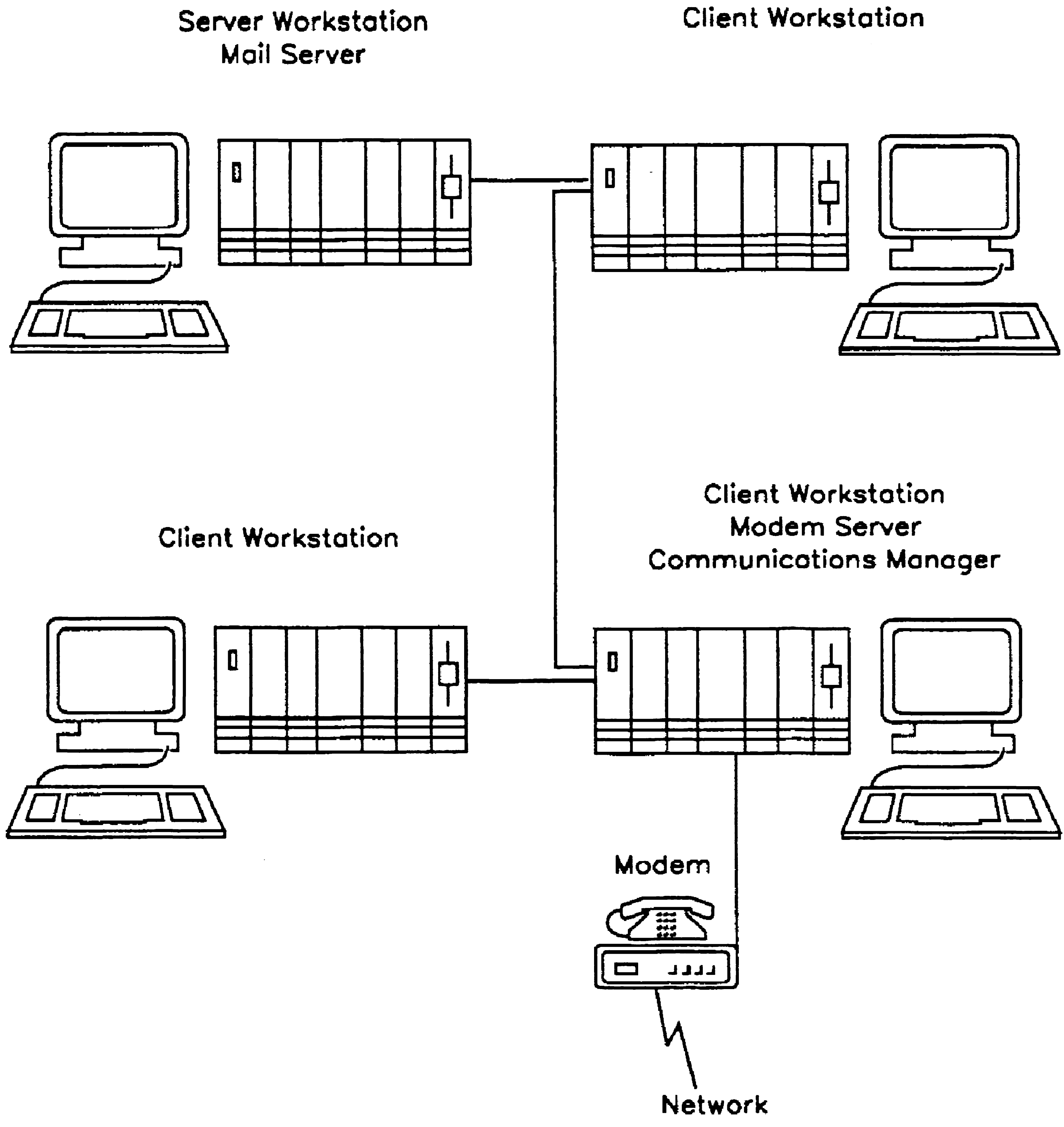
The following system services are included with electronic mail applications. They are frequently installed and configured by the system administrator.

Mail Service	This system service controls distribution of incoming and outgoing mail for the entire cluster. It must be installed on the server before the Modem Service and Communications Manager are installed.
Modem Service	This system service is required to use a modem. It is installed on the system where the modem is connected.
Communications Manager	This system service provides communications between mail centers. It is installed after the Modem Service on the system to which the communication line is connected.

Figure 9-2 shows an example of the electronic mail services installed within a cluster. See the documentation for your mail system for more detailed information.



Figure 9-2. Electronic Mail System Services in a Cluster



603.9-2

### Network System Services

Network software is also installed and maintained by the system administrator. It may include a number of different system services, such as BNet, which provides communications among remote clusters, and one or more media system services, which provide data communications protocols. The types of network services you install are determined by the physical data links that connect your cluster to other nodes in the network.

See the manuals for your networking products for more detailed information.

### Other System Services

Many other system services are available as separate products or as part of application software packages. These include the following:

- Clustering services, for allowing IBM PC®s and other compatible computers to share resources on a CTOS cluster
- Indexed Sequential Access Method (ISAM) Service, for database applications
- Communications services, such as SNA and X.25 gateways, for communicating with external data networks

See the product documentation for information about installing them.

# Where to Install System Services

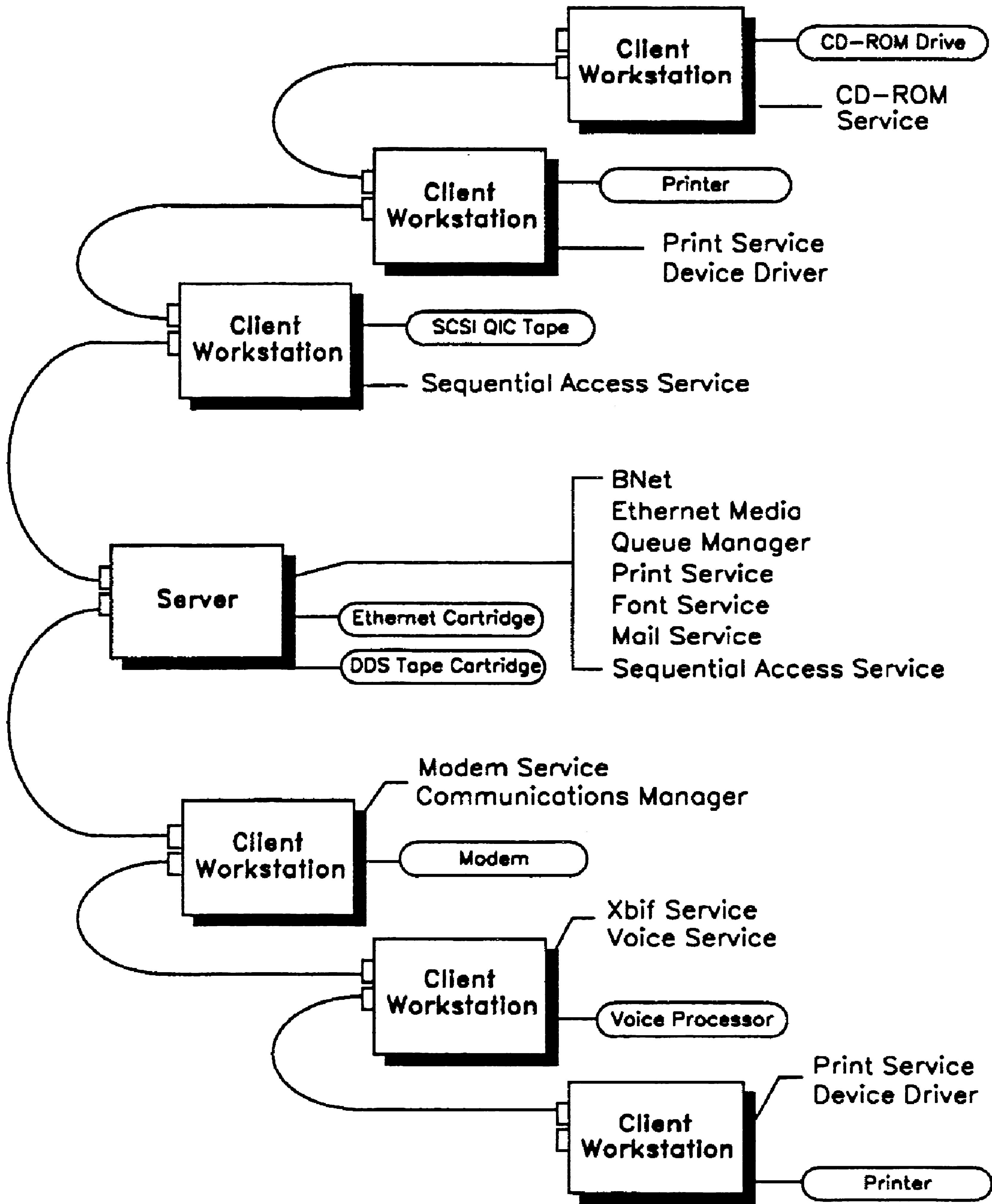
It is not unusual for many system services to be installed within a cluster, as shown in Figure 9-3. In general, system services are installed on the workstation or SRP processor to which the resource is attached. There are exceptions to this, however; for example, the Queue Manager and Mail Service are always installed on the server. See “What System Services Do You Need?” earlier in this section, for general information. Always read the release documentation and operating guides for the most current information.

Where you locate resources, and subsequently install system services, depends on several factors. The layout of your work area can be a factor in where you set up equipment. For example, a modem must be installed near a telephone line and a printer requires adequate space.

For shared system services, such as printing, you need to consider the type of work for which a workstation or processor is used. For example, if a workstation is often used for graphics and font applications, the printing services might decrease performance of other applications. Similarly, if an SRP processor is used extensively for communications applications, the printing services might perform more efficiently on a different processor.

The following section, “Calculating Memory Requirements,” helps you determine where to install system services. See also Section 15, “Optimizing System Performance.”

Figure 9-3. System Services Installed Throughout a Cluster



## Calculating Memory Requirements

The term *memory* refers to random access memory (RAM), which is located on workstations, IOPs, and SRP processors. The term *memory* does not refer to disk storage space. All workstations, even those that are diskless, contain memory.

System services and applications require memory in which to execute. The way you determine memory requirements for applications and system services is described below.

### Operating Systems Without Virtual Memory

Processors running real-mode and protected-mode operating systems without virtual memory use *partitioned memory management*. On such processors, system services and applications are loaded into individual memory partitions, and the amount of memory available at any given time is the total amount of processor memory minus the amount of memory allocated to those individual partitions.

### Virtual Memory Operating Systems

Processors running virtual memory operating systems use *demand paging*. On such processors, no single amount of memory is specifically available or unavailable at any given time. Instead of being loaded into partitions, applications and system services are circulated in and out of memory in 4K-byte pages. Therefore, the number of programs and system services that can be installed and active on a virtual memory system is not static.

### How Much Memory Is Available?

To determine the amount of memory available on a partitioned memory system, you use the **Partition Status** command. The **Partition Status** display is pictured in Figure 9-4; instructions for using **Partition Status** are included later in this section.

**Figure 9-4. Partition Status Display**

---

Partition	Allocated	Used	Run file executing
System	856K	856K	pClstrLfs 3.4
2 Vdm_VGA 3.4	184K	182K	
3 Mouse Server	21K	19K	
4 XvtChService	337K	335K	
5 Primary	591K	230K	SystemMgr.run
Total RAM:	4096K		Available: 2747K

---

The **Partition Status** display lists the amount of memory allocated for each application or system service partition. Memory statistics are displayed in K bytes; each K byte is equivalent to 1024 bytes of memory.

To determine how much memory remains available on the processor, add all values in the *Allocated* column of the **Partition Status** display and subtract that total from the total amount of processor RAM. For example, in Figure 9-4, the amount of allocated memory equals 1989K bytes (856+184+21+337+591=1989). The amount of processor memory, as displayed in the *Total RAM* field, is 4096K bytes. Therefore, the amount of unallocated memory that remains available on the processor is 2107K bytes (4096-1989=2107).

**Note:** *The amount of memory displayed in the Available field of the Partition Status display includes unusable fragments.*

On virtual memory systems, the **Partition Status** command displays the amount of memory available in frames. For the most part, you do not need to be concerned with calculating available memory on virtual memory systems; however, a small number of available frames may indicate that the system is not performing optimally. See Section 15, "Optimizing System Performance," for information about optimizing the use of virtual memory.

See the *CTOS Executive Reference Manual* for more detailed information about the **Partition Status** display.



### Starting Partition Status on a Single-Processor Workstation

To start **Partition Status** on a single-processor workstation, follow these steps:

1. On the Executive command line, type **Partition Status**.
2. Press **GO**.

### Starting Partition Status on a Multiprocessor System

To start the **Partition Status** command on the master processor of a multiprocessor workstation, simply type **Partition Status**, and then press **GO**.

To start the **Partition Status** command on an IOP or on an SRP processor, start a Cluster View session on the appropriate processor; then execute the **Partition Status** command from the Executive command line.

For information about using Cluster View, see Section 4, "Using Administrative Tools."

### How Much Memory Is Required?

The following factors determine the amount of memory that is required to install the system services you need on a partitioned memory system. These factors must be considered separately for each processor:

- What is the total amount of memory (RAM) on the processor? (See "How Much Memory Is Available?," above.)
- How much memory is required for the operating system and video service partitions? (See the Software Release Announcements.)
- How much memory is required for all the system service partitions? (See the release documentation for your software packages.)
- How much memory is required for the largest application partition? (See the release documentation for the applications you plan to use.)

To calculate the total memory requirement, add the above factors, as shown in the following example:

Operating system partition	856K
Video service partition	184K
System services partitions	458K
Partition for the largest application	+800K
Total	<u>2298K</u>

The total memory requirement for the operating system, system services, and largest application partition must not exceed the total amount of processor memory (RAM).

On virtual memory systems, the paging service moves programs and data in and out of memory as they are needed. Therefore, you can install more system services and start more applications than actually fit into memory on the processor, based on the *working set* of memory that is required by the system service or application. A working set is the amount of memory required by an application to function with reasonable performance. See Section 15, "Optimizing System Performance," for more detailed information about virtual memory management, the paging service, and working set values.

## Installing System Services on a Single-Processor Workstation

You can install system services with Executive commands or during system initialization. Table 9-1 lists the Executive commands and run-file names for the system services supplied with Standard Software.

See the *CTOS Executive Reference Manual* for information about parameter values. See the release documentation for information about installing system services for other applications.

## Installing From the Executive

You can install system services with Executive commands. This method is generally used in the following circumstances:

- When you install a system service for the first time and want to make sure that it works as you expect
- When you need a system service only at certain predictable times, for example, if you use the Sequential Access Service once a day for backups

**Note:** *System services cannot be installed while Context Manager is running on the workstation.*

To install a system service with the Executive, follow these steps:

1. Type the command name on the Executive command line; then press RETURN (see Table 9-1).
2. Fill in parameter fields as required; see the appropriate manual or release documentation.
3. Press GO.

**Table 9-1. System Service Commands and Run Files**

<b>System Service</b>	<b>Executive Command</b>	<b>Run File</b>
CD-ROM Service	<b>Install CDROM Service</b>	<i>CDROMService.run</i>
Cluster File Access File Filter	<b>Install CFA File Filter</b>	<i>CfaFf.run</i>
Cluster File Access Server Service	<b>Install CFA Server Service</b>	<i>CfaM.run</i>
Cluster File Access Workstation Agent	<b>Install CFA Workstation Agent</b>	<i>CfaWa.run</i>
Command Access Service	<b>Install Command Access Service</b>	<i>AccessService.run</i>
DataComm Service	<b>Install DataComm Service</b>	<i>DcxService.run</i>

continued

## Installing System Services

---

Table 9-1. System Service Commands and Run Files (cont.)

System Service	Executive Command	Run File
Math Service	<b>Install Math Service</b>	<i>MathService.run</i>
MCR Service	<b>Install MCR Service</b>	<i>MCRService.run</i>
Mouse Service	<b>Install Mouse Service</b>	<i>Mouse.run</i>
Queue Manager	<b>Install Queue Manager</b>	<i>InstallQMgr.run</i>
Remote Keyboard Video Service	None	<i>RKVS.run</i>
Remote User Manager	None	<i>RUM.run</i>
Screen Print Service	<b>Install Screen Print</b>	<i>InstallScreenPrint.run</i>
Sequential AccessService	<b>Install Sequential Access Service</b>	<i>InstallSeqService.run</i>
Spooler	<b>Install Spooler</b>	<i>Spooler.run</i>
Statistics Service	<b>Install Statistics Service</b>	<i>Statistics.run</i>
Voice Service	<b>Install Voice Service</b>	<i>AudioService.run</i>
XBIF Service	<b>Install XBIF Service</b>	<i>Xbif.run</i>
XC-002 Service	<b>Install XC002 Service</b>	<i>XC002Service.run</i>

## Installing During System Initialization

System services that you use all the time, such as those for printing or electronic mail, can be installed during system initialization. This is convenient and automatic; whenever you reboot the workstation, system services are automatically installed.

After the workstation bootstraps, the system initialization program searches for a file named *[Sys]<Sys>SysInit.jcl*. If it exists, its contents are read and executed. You create *SysInit.jcl* to install the appropriate system services on the workstation. Figure 9-5 shows a system initialization file for a single-processor workstation.

**Figure 9-5. System Initialization File for a Single-Processor Workstation**

---

```
Job SysInit
ContinueOnError
Run [Sys]<Sys>Xbif.run
Run [Sys]<Sys>InstallQMgr.run,y,30
Run [Sys]<Gps>GpsInstall.run
Run [Sys]<Sys>MailService.run
Run [Sys]<Sys>ModemServer.run
Run [Sys]<Sys>CommunicationsManager.run,Lin1
Run [Sys]<Sys>InstallSeqService.run,[QIC]
Run [Sys]<Sys>MouseService.run
End
```

---

### Creating a System Initialization File

To create a system initialization file, you use the Editor to write a simple program in Job Control Language (JCL). Table 9-2 lists the JCL statements you use in this file. See “JCL Syntax,” later in this section, for rules about punctuation and spacing. See Section 4, “Using Administrative Tools,” for information about using the Editor.

After you create the file, reboot the workstation to install the system services.

### Workstations That Boot From the Server

You can create system initialization files for client workstations that boot from the server. Such files must be located in *[Sys]<Sys>* on the server and are named as follows:

*[Sys]<Sys>WsNNN>SysInit.jcl*

where NNN is the three-digit workstation-type number or, if that file does not exist for a particular workstation type:

*[Sys]<Sys>Ws>SysInit.jcl*

If neither *WsNNN>SysInit.jcl* nor *Ws>SysInit.jcl* exists, no system initialization sequence is executed on the workstation.



See also “Workstation-Type Numbers,” in Section 5, “Bootstrapping,” for more information about the system initialization sequence.

You can also create system initialization files that correspond to workstation hardware IDs. See Section 5, “Bootstrapping,” for more detailed information.

**Table 9-2. JCL Statements for System Initialization Files**

Statement	Description
Command	This statement specifies the name of an Executive command. To be executed during system initialization, the command must be present in <i>[Sys]&lt;Sys&gt;Sys.cmds</i> .
ContinueOnError	This statement forces continuation of the system initialization file, even if an error occurs. It affects statements that follow it until the end of the file, or until a <i>CancelOnError</i> statement occurs.
CancelOnError	This statement reinstates the <i>CancelOnError</i> condition, which terminates system initialization when an error occurs.
End	This statement defines the end of a Batch job. End your system initialization file with it, as shown in Figure 9-5.
FrontPanel (for SRPs only)	This statement displays the two-digit number of your choice on the front panel. Use it to mark the progress of the system initialization sequence. This can be helpful for isolating problems.
Job	This statement defines a name for the batch job. For system initialization, specify <b>SysInit</b> , as shown in Figure 9-5.
Run	This statement specifies the name of a run file to be executed.
RunNoWait (for multiprocessor systems only)	This statement instructs the master processor to execute the next statement in the system initialization file immediately. See also “Using the RunNoWait JCL Statement,” later in this section.
<i>xPnn</i> (for multiprocessor systems only)	This statement specifies the processor on which to install the system service, where <i>xPnn</i> is the four-character processor ID.



## Installing System Services on a Multiprocessor Workstation

Before you install system services on a multiprocessor workstation, keep in mind the following points:

- Some system services must be installed on a certain type of IOP.
- Some system services must be installed on the processor that is physically connected to the resource it is going to manage.
- Some system services must be installed before or after other system services.

If the multiprocessor workstation is equipped with a keyboard and monitor, you can install system services directly onto the master processor with the Executive; see “Installing System Services on a Single-Processor Workstation,” earlier in this section. You then use Cluster View to install system services onto the IOPs, as described later in this section.

If the multiprocessor workstation does not have a keyboard and monitor, you use Cluster View to install system services from a client workstation onto the master processor, as well as the IOPs.

### Installing With Cluster View

You can use Cluster View to install system services on IOPs (and the master processor, on a videoless server) with Executive commands. This is a good technique when you are setting up a multiprocessor workstation for the first time or are adding a system service after installing a new application. After you have determined that the system services install correctly, you can add them to the system initialization file.

**Note:** *If the Remote User Manager is running on a processor, use the Administrator Cluster View command (rather than the Cluster View command) to install system services.*

## Installing System Services

---

To install a system service with Cluster View, follow these steps:

1. On the Executive command line, type **Administrator Cluster View**; then press **RETURN**.
2. Fill in the command form, as shown in the following example. (See Table 4-3 for a description of parameter fields.)

Administrator Cluster View

[Processor name]	EP01
[User name]	Admin
[User file password]	####
[Node name]	
[Old XE run file?]	
[Run file to invoke]	
[Partition size]	

3. Press **GO**.
4. If necessary, start the Executive by signing on or exiting the application that is running.
5. Execute the command for the system service you want to install (see Table 9-1).

## Installing During System Initialization

In most cases, a multiprocessor workstation runs many system services, so it is convenient to install them during system initialization. Figure 9-6 shows a sample system initialization file for a multiprocessor workstation.

To create a system initialization file for a multiprocessor system, you use the Editor application. See Section 4, "Using Administrative Tools," for information about using the Editor.

---

**Figure 9-6. System Initialization File for a Multiprocessor Workstation**

---

```
Job SysInit
;Example file for a multiprocessor workstation

EP00
;This is the master processor
Run [Sys]<Sys>FontService.run, [Sys]<Gps>Font.dbs,11264
Run [Sys]<Sys>InstallQMgr.run,y,20
Run [Sys]<Gps>GpsInstall.run
Run [Sys]<Sys>InstallSeqService.run, [QIC]
Run [Sys]<Sys>ProtocolManager.run

EP01
;This is a Communications IOP
Run [Sys]<Sys>BMZip.run
RunNoWait [Sys]<Sys>RUM.run

EP02
;This is an Ethernet IOP
Run [Sys]<Sys>Install802/3.run
Run [Sys]<Sys>InstallTransnet.run,,,,,,,,,EDL
Run [Sys]<Sys>BotlConfig.run,,,EDL
Run [Sys]<Sys>BNetSession.run
Run [Sys]<Sys>BNet.run
Run [Sys]<Sys>NS.run
Run [Sys]<Sys>MailService.run
RunNoWait [Sys]<Sys>RUM.run

EP00
;This is the master processor again
RunNoWait [Sys]<Sys>RUM.run

End
```

---

### JCL Statements for Multiprocessor Systems

Table 9-2 lists JCL statements for multiprocessor systems. Special considerations for the processor ID, RunNoWait statement, and installation of the Remote User Manager are described later in this section.

JCL syntax is the same as for single-processors workstations. See “JCL Syntax,” later in this section.

### Specifying a Processor

On a multiprocessor workstation, the master processor controls execution of the system initialization file. The master processor recognizes IDs for the other processors, so that system services can be installed on the IOPs.

To specify a processor, you type the four-character ID for the processor, for example EP01. Statements that follow the processor ID are executed on that processor until another processor is identified; see Figure 9-6 for an example. For a description of processor IDs, see “CTOS I/O Processor Cards,” in Section 2, “Understanding Hardware.”

## Installing System Services on an SRP

Considerations and instructions for installing system services on SRPs are essentially the same as those described for multiprocessor workstations, earlier in this section.

You can use Cluster View to install system services with Executive commands, or you can install system services during system initialization. See “Installing System Services on a Multiprocessor Workstation,” for information about both of those methods.

A sample system initialization file for an SRP is shown in Figure 9-7. See Table 9-2 for a description of JCL statements.

Figure 9-7. System Initialization File for an SRP

---

```
Job InitSRP
;Example SysInit.jcl file for an SRP

FrontPanel 99
GP01
Run [Sys]<Sys>Net.run,Ranger,74,4,120,,64,1500,6

FrontPanel 98
CP00
Run [Sys]<Sys>ModemService.run,[Sys]<Sys>ModemService.config
Run [Sys]<Sys>Async.run,[Sys]<Sys>AsyncConfig.sys

FrontPanel 97
TP01
Run [Sys]<Sys>Hdlc.run,[Sys]<Sys>HdlcConfig.sys

FrontPanel 96
GP01
Run [Sys]<Sys>ProtocolManager.run
Run [Sys]<Sys>Install802/3.run
Run [Sys]<Sys>InstallTransNet.run,,,,,,,,,Edl
Run [Sys]<Sys>BotlConfig.run,,,Edl
Run [Sys]<Sys>BNetSession.run,[Sys]<Sys>BnetSessionConfig.sys
Run [Sys]<Sys>BNet.run,[Sys]<Sys>BnetConfig.sys
Run [Sys]<Sys>NAC.run,[Sys]<Sys>Netcfg.sys
Run [Sys]<Sys>NS.run
RunNoWait [Sys]<Sys>RUM.run

FrontPanel 95
GP00
Run [Sys]<Sys>FontService.run
Run [Sys]<Sys>InstallQMgr.run,y,20
Run [Sys]<GPS>GpsInstall.run
Run [Sys]<Sys>MailService.run
Run [Sys]<Sys>InstallSeqService.run,[QIC]
RunNoWait [Sys]<UTE>UniScopeGateway.run,1

FrontPanel 94
GP00
Run [Sys]<Sys>XVTService.run
RunNoWait [Sys]<Sys>RUM.run

End
```

---



### Specifying a Processor

On an SRP, the master processor controls execution of the system initialization file. The master processor recognizes IDs for the other processors so that system services can be installed anywhere on the SRP.

To specify a processor, you type the four-character ID for the processor, for example GP00, FP01, and so on. Statements that follow the processor ID are executed on that processor until another processor is identified; see Figure 9-7 for an example. For a description of processor IDs, see “Processor Boards,” in Section 2, “Understanding Hardware.”

### Using Keyswitch Files

When an SRP is bootstrapped, the keyswitch on the front panel is set to one of three positions: N, M, or R. That keyswitch position determines the system initialization file that is executed after the SRP bootstraps. For more information, see “Keyswitch Positions,” in Section 5, “Bootstrapping.”

The keyswitch-controlled system initialization files are named as follows:

*[Sys]<Sys>SysInit.k.jcl*

where *k* corresponds to the keyswitch position.

If a system initialization file for a particular keyswitch does not exist, *[Sys]<Sys>SysInit.jcl* is used instead. No default system initialization file is installed with Standard Software; you must create your own system initialization files.

## JCL Syntax

When you create a system initialization file, you are writing a program in Job Control Language. (JCL files are processed by the Batch facility, which can be used for jobs other than system initialization.) The following sections describe syntax rules that apply to system initialization. See the *CTOS Batch Manager II Installation, Configuration, and Programming Guide* for more detailed information.



## Specifying Run Files and Command Names

Separate the name of a run file or command from the Run or Command statement with at least one space or tab, as shown in the following examples:

### Run file

```
Run [Sys]<Sys>MouseService.run
```

### Command

```
Command Install Mouse Service
```

## Specifying Parameters

When a run file or command requires parameters, separate them with commas, as shown in the following examples:

### Run file

```
Run [Sys]<Sys>InstallQMgr.run, yes, 10
```

### Command

```
Command Install Queue Manager, yes, 10
```

The area between two commas is equivalent to an Executive command form field. If you want to leave a parameter blank, you must enter commas as a place holder for the field. The following example shows three blank fields between the parameter values 75 and 50:

```
Run [Sys]<Sys>Net.run, SJ-Node, 67, 2, 75,,,,50
```

You do not need to enter commas, however, to omit parameters at the end of a statement.

### Specifying Subparameters

When a parameter consists of more than one value per field, enclose subparameters in parentheses, as shown in the following example:

```
Command Disable Cache, (File1, File2), yes
```

### Entering Multiple-Line Parameters

When parameters exceed a single line, use an ampersand (&) to indicate that the statement continues on the next line, as shown in the following example:

```
Run [Sys]<Sys>CommunicationsManager.run, &  
VeryLongNameIsCarriedOver
```

### Entering Comments

To enter a comment, precede text with a semicolon (;), as shown in the following example:

```
;This is a comment
```

You can add comments to clarify the contents of a file (for yourself or others who might read it) or to temporarily disable a JCL statement. The following example shows both:

```
Job SysInit  
ContinueOnError  
Run [Sys]<Sys>Xbif.run  
;Seq Service disabled.  
;Run [Sys]<Sys>InstallSeqService.run, [QIC]  
Run [Sys]<Sys>MouseService.run  
Run [Sys]<Sys>AudioService.run  
End
```

## Installing the Remote User Manager

The Remote User Manager is an optional system service for use on protected-mode processors. (See “Cluster View,” in Section 4, “Using Administrative Tools.”) After it is installed, no other system services can be installed on the processor. Therefore, the Remote User Manager must be installed properly; otherwise, the system initialization sequence stops and the system appears to be frozen.

On a single-processor server or a master processor, the Remote User Manager must be the final system service to be installed during system initialization. Figure 9-7 shows how several system services are initially installed on GP00 (the master processor) of an SRP; then, after all other system services are installed on the other processors, the Remote User Manager (*RUM.run*) is installed on the master processor.

When installing the Remote User Manager on a processor other than the master processor, it must be the last system service installed on that processor. Figure 9-7 shows how the Remote User Manager is installed as the last system service on GP01.

*Use the RunNoWait statement to install the Remote User Manager.* If you inadvertently use the Run statement, the Remote User Manager is installed, but the rest of the system initialization file cannot be executed.

## Using the RunNoWait JCL Statement

The RunNoWait statement works differently from the other system initialization JCL statements. Although it is not necessary to understand what it does to install the Remote User Manager, you might be interested in the difference between it and the other statements. Also, if you are using any customized system services (those that have been changed from the released versions), you may need to know when to use the RunNoWait statement.

As explained earlier, the master processor controls the system initialization procedure. When it encounters Run or Command statements, it executes them one after another, in the order they appear in the system initialization file. For example, in Figure 9-7, first the Font Service is installed on GP00, then the Queue Manager, and so on. During this procedure, the master processor waits for confirmation that a statement has finished executing before it starts the next statement. In most cases, this method works well and system initialization proceeds in an orderly fashion.

To do its job, however, the Remote User Manager must remain active on the processor on which it is installed. Because it does not finish executing, the master processor does not receive a message to continue with system initialization. Therefore, a special statement, RunNoWait, is required to install the Remote User Manager. When the master processor encounters the RunNoWait statement, it does not wait for that statement to complete before continuing with the next statement in the file. So, system initialization continues, even though the Remote User Manager never finishes executing. If you inadvertently install the Remote User Manager with the Run statement, the system initialization procedure waits indefinitely for the Remote User Manager to finish executing and the system appears to be frozen.

Although the major purpose of the RunNoWait statement is to install the Remote User Manager, it can also be used to install system services on multiple processors simultaneously. Theoretically, this can speed up system initialization, but it can also cause problems if system services must be installed in a particular order. Also, when you use the RunNoWait statement, the master processor attempts to read several files at the same time. If all the files are located on the system volume, the resulting disk activity can actually slow down the system initialization procedure. Therefore, it is usually better to reserve RunNoWait for situations where it is required, such as installing the Remote User Manager.

---

## Common Problems With System Services

Table 9-3 describes several common errors that are associated with system services.

Table 9-3. System Service Errors

---

Error Code	Description
31	<p>No such request</p> <p>This error occurs when a loadable request (as described in Section 8) is not available to a system service. It can happen when a workstation or SRP requires rebooting after an application has been installed or updated. To correct it, try rebooting the system. This error also occurs if loadable requests were not properly merged during software installation. Try repeating the software installation procedure by using the <b>Installation Manager</b> command.</p>
33	<p>Service not available</p> <p>This error occurs when you execute a command requiring a system service that has not been installed. For example, before you can use a mouse, you must execute the <b>Install Mouse Service</b> command to install Mouse Services on your workstation. See the documentation for the application you are trying to use.</p>
203	<p>No such file</p> <p>This error can occur at the beginning of system initialization if the run file <code>[Sys]&lt;Sys&gt;Batch.run</code> is not present on the system (also <code>[Sys]&lt;Sys&gt;CLI.run</code> on SRPs only). Use the <b>Files</b> command to make sure that it is there. If it is not, copy it from the system utilities distribution media.</p>

---





## Section 10

# Accessing Data Throughout the Cluster

## Using Disks on the Server

A built-in feature of the CTOS operating system allows client-workstation users to access disks located on the server. To do so, you simply specify an exclamation point (!) in front of the volume or device name of the disk you want to access, as shown in the following example:

```
[[d1]<Dept>FileOnServer.txt
```

## Accessing Disks on Client Workstations

It is also possible to access files located on workstations throughout the cluster. The Cluster File Access (CFA) system services provide selective access to disks from client workstation to client workstation. A configuration file on each workstation designates the availability of its hard disks to other workstations in the cluster.

*Note:* Cluster File Access does not work in clusters with an SRP or EISA server.

Cluster File Access consists of the following system services:

- The CFA Server Service is installed on the server to enable Cluster File Access.
- The CFA File Filter is installed on workstations that will be accessing disks on other client workstations.
- The CFA Workstation Agent provides access to disks on the workstation where it is installed.

A client workstation does not need to run both the CFA File Filter and the CFA Workstation Agent. For example, a workstation running the CFA File Filter only can have access to other disks without providing access to its own. Only those disks on workstations running the CFA Workstation Agent are accessible throughout the cluster.

### Configuring Cluster File Access

Before you install the Cluster File Access system services for the first time, you create a configuration file by using the **CFA Configure** command on the server and on each client workstation.

To use the **CFA Configure** command, follow these steps:

1. On the Executive command line, type **CFA Configure**; then press **RETURN**.
2. In the command form, specify a name for the configuration file. The default is *[d0]<Sys>CFAConfig.sys*.
3. Press **GO**. The CFA Cluster Workstation Configurator appears, as shown in Figure 10-1.
4. Press **LEFT ARROW** or **RIGHT ARROW** to select an access level for each disk. (See "Protecting Directories," in Section 6, "Implementing System Security," for a description of read and modify access levels.)
5. Press **FINISH**, then **GO**, to save the configuration file.

Figure 10-1. CFA Configure Display

CFA Cluster Workstation Configurator				
Disk	Volume Name	Access		
D0	CPG-0	Read	Modify	None
D1	CPG-1	Read	Modify	None
F0	F0	Read	Modify	None

603.10-1

## Installing the CFA Server Service

The Cluster File Access Server Service is installed on the server workstation only. To install it, type **Install CFA Server Service** on the Executive command line; then press **GO**.

Alternatively, you can install the Cluster File Access Server Service during system initialization. To do so, include the following entry in the system initialization JCL file:

**Run [Sys]<Sys>CfaM.run**

## Installing the CFA Workstation Agent

The Cluster File Access Workstation Agent is installed on each client workstation that has disks to be accessed by other workstations in the cluster. To install it, follow these steps:

1. On the Executive command line, type **Install CFA Workstation Agent**; then press **RETURN**.
2. Fill in the command form. Parameter fields are shown below and described in Table 10-1.

```
Install CFA Workstation Agent
  [Config file ([d0]<Sys>CFAConfig.sys)] _____
  [Maximum remote clients]                _____
```

3. Press **GO**.

**Table 10-1. Install CFA Workstation Agent Parameters**

Field Name	Description
<i>[Config file]</i>	Default: <i>[d0]&lt;Sys&gt;CFAConfig.sys</i>  Enter the file specification of the configuration file containing the disk access information you want to use. Access levels are assigned to disks with the <b>CFA Configure</b> command.
<i>[Maximum remote clients]</i>	Default: 32  Enter the maximum number of applications running elsewhere in the cluster that can access local disks simultaneously, via the CFA Workstation Agent.

## Accessing Data Throughout the Cluster

---

Alternatively, you can install the Cluster File Access Workstation Agent during system initialization. To do so, include the following entry in the system initialization JCL file:

**Run [Sys]<Sys>CfaWA.run, ConfigFile, RemoteClients**

## Installing the CFA File Filter

The Cluster File Access File Filter is installed on each client workstation that will be accessing disks on other client workstations. To install it, type **Install CFA File Filter** on the Executive command line; then press **GO**.

Alternatively, you can install the Cluster File Access File Filter during system initialization. To do so, include the following entry in the system initialization JCL file:

**Run [Sys]<Sys>CfaFF.run**

## Using Cluster File Access

To use a disk configured for Cluster File Access, specify its unique volume name (rather than its device name) in file specifications or command forms, as shown in the following example:

```
Files  
[File list (*)] [TechPubs]<Sys>*
```

---

**Note:** *For all disks in the cluster to be accessible via Cluster File Access, each must have a unique volume name.*

### Accessing Floppy Disk Drives

To access a floppy disk, specify its device specification followed by the volume name of the system volume. Separate the floppy disk drive specification and the system volume name with a colon, as shown in the following example:

```
Files  
[File list (*)] [f0:TechPubs]<Sys>*
```

---

If the system volume on the workstation you want to access has not been configured for CFA modify access, you will not be able to access its floppy disk drive. In addition, if the system volume has a password, that password may be required to access the floppy disk drive. See the **CFA Configure** command in the *CTOS Executive Reference Manual* for more information.

### Displaying CFA Volume Names

To display the volume names of disks available in your cluster, use the **CFA Display Volume Information** command. To do so, type **CFA Display Volume Information** on the Executive command line; then press GO.

See the *CTOS Executive Reference Manual* for more detailed information about the **CFA Display Volume Information** command.

## Accessing Additional Resources

As an alternative to Cluster File Access, the BNet Cluster Access Services allow you to access files and other resources throughout the cluster and on other network nodes. BNet Cluster Access is included with CTOS as a separately installable software package. For more information, see the manual and release documentation for it.





# Section 11

## Adding Hard Disks

### What Is a CTOS Volume?

Many systems, both workstations and SRPs, contain more than one hard disk for data storage. When you set up a new system, you initialize the system volume during the software installation procedure. After the system volume has been initialized and you have installed Standard Software, you initialize the other disks on the system. A disk must be initialized to be recognized by the CTOS operating system as a *valid volume*.

This section describes how to initialize disks with the **Format Disk** command. It also describes how to initialize a partitioned disk for use with other operating systems. See also the following manuals:

- For information about initializing a system volume during software installation, see the *CTOS System Software Installation Guide*.
- For information about initializing floppy diskettes, see the *CTOS Executive User's Guide*.

### Workstation Disks

For X-Bus and X-Bus+ workstations, disks are supplied in add-on modules and cartridges. The model of disk you use depends on your workstation and processor type. See the workstation hardware guides for installation instructions.

EISA/ISA workstations are usually equipped with one or more internally housed disks (often called *drives*). For information about installing hard disks into EISA/ISA workstations, see the appropriate hardware installation guide.

### Workstation Disk Compatibility

Different types of disks are compatible with different workstation models, as described below:

Non-SCSI disks	X-Bus workstations only
SCSI disks	X-Bus, X-Bus+, and EISA/ISA workstations
IDE drives	EISA/ISA workstations only

Recognition of a disk on any system depends on a number of factors, such as compatibility of the controller and placement of the drive. See your workstation hardware installation guide for more specific information.

### Device Names for Workstation Hard Disks

On workstations, hard disk devices are named  $dn$ , where  $n$  indicates the placement of the disk drive. Disk device names can change when you add a disk to the workstation. For example, if you add a new disk module between two existing disk modules on an X-Bus workstation, the device name of the third disk changes accordingly. A volume name, however, which you define when you initialize the disk, remains the same when physical placement of the disk is changed.

On X-Bus workstations, the first hard disk to the right of the processor is  $d0$ , the second,  $d1$ , and so on.

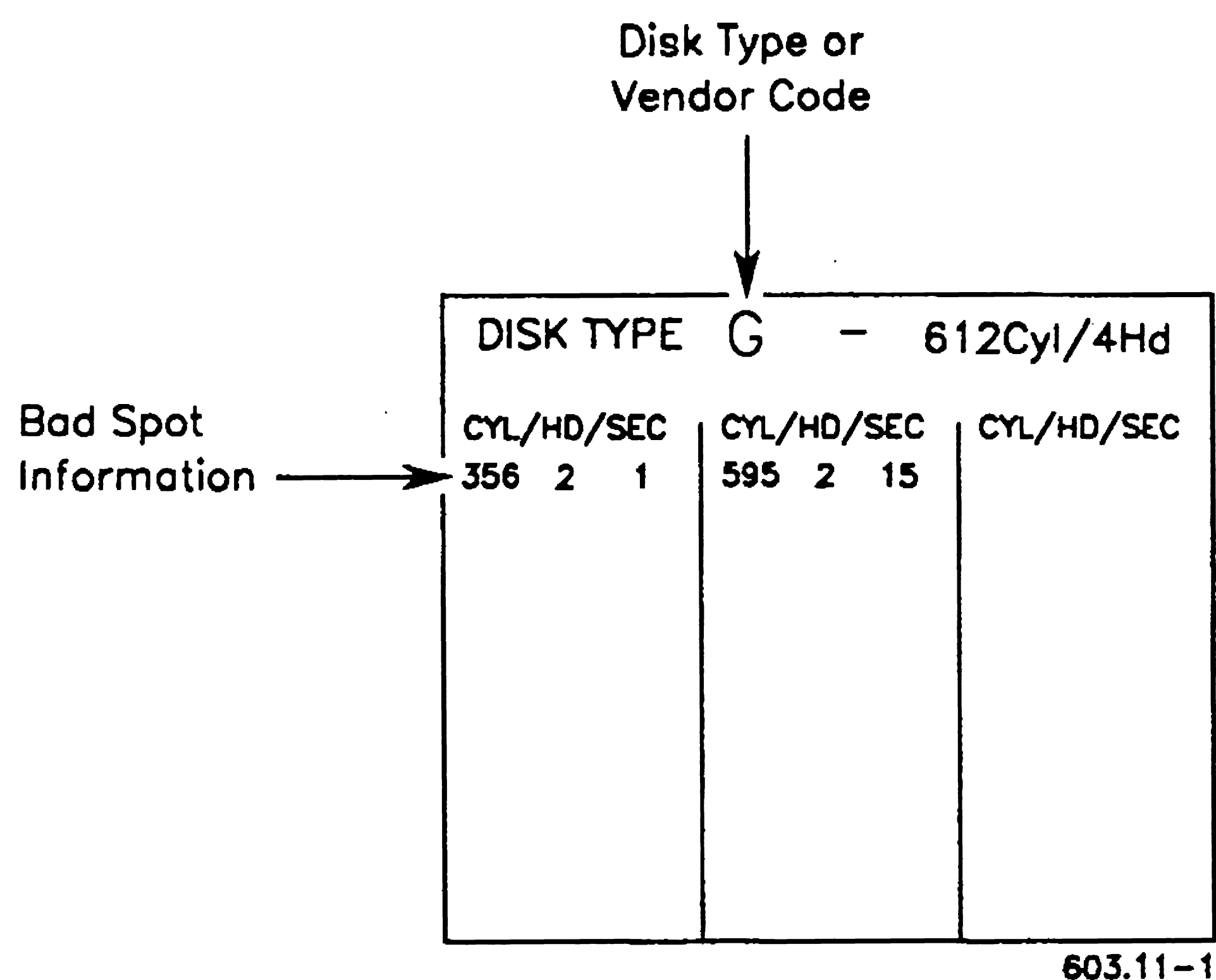
On X-Bus+ workstations, hard disks are numbered  $d0$  and  $d1$ , from top to bottom, in the base unit. Disk numbering then continues sequentially in the SCSI expansion unit to the left of the base unit, followed by the X-Bus to the right of the base unit. See the *SuperGen Series 5000 Hardware Installation and Maintenance Guide* for detailed information about device names.

On EISA/ISA workstations, hard disks are named  $d0$ ,  $d1$ , and so on.

## Disk Type and Bad Spot Report

X-Bus disk modules are labeled on the bottom with a disk type (sometimes called vendor code), as shown in Figure 11-1. In addition, non-SCSI disk modules are labeled with a bad spot report, also shown in Figure 11-1. In situations described later in this section, you may need to use the disk-type code and bad spot report.

Figure 11-1. Workstation Disk Type and Bad Spot Report



## Partitioned Disks

On EISA/ISA workstations, you can format a disk so that it contains more than one partition. One partition can be used for CTOS files and another for non-CTOS files, such as those associated with DOS and its applications.

See "Formatting a Partitioned Disk," later in this section, and the **Format Disk** command in the *CTOS Executive Reference Manual*, for more detailed information.

# SRP Disks

Many sizes and types of disks are available for the SRP. They can be installed within an SRP cabinet, or they can be external to the system. In some cases, the system administrator is responsible for installing disks; in other cases, disks are installed by a field service engineer. If you need information about installing disks, see the installation guide for your SRP.

## SRP Disk Compatibility

On an SRP, different types of disks are compatible with different processors, as described below:

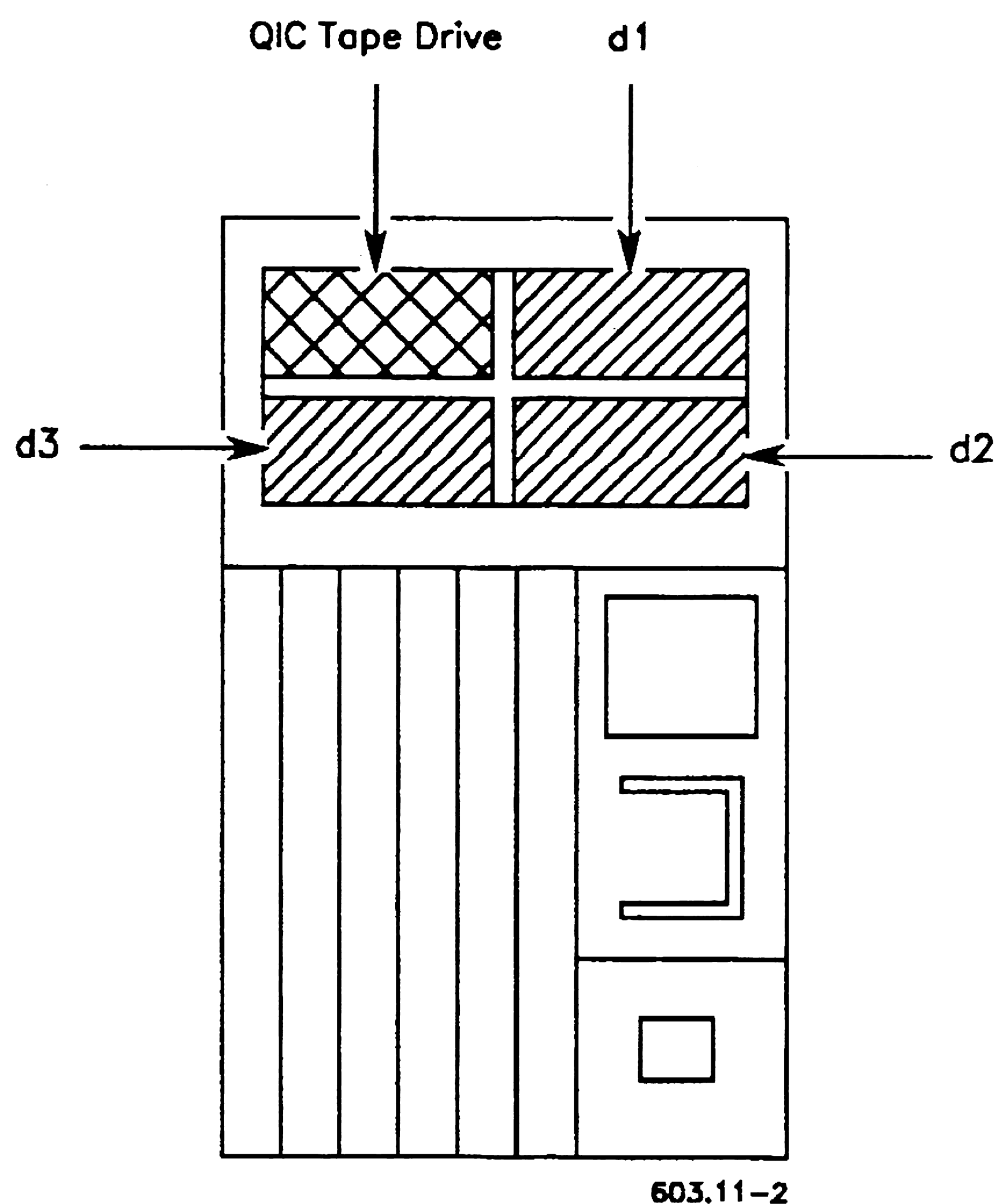
- SCSI disks are connected to a General Processor with SCSI Interface (GP+SI).
- Non-SCSI ST-506 disks are connected to a File Processor (FP).
- Non-SCSI SMD disks are connected to a Data Processor (DP).

## Device Names for SRP Disk Drives

The operating system assigns default device names for disk drives controlled by the master processor only. If the master processor is a GP+SI, default disk device names are *d1* through *d15* (see the *XE-530 Shared Resource Processor Hardware Installation Guide*). On an FP master processor, defaults are *d1* through *d3*. Because a QIC tape drive occupies the first drive slot (as shown in Figure 11-2), there is no disk drive named *d0* on an SRP. If the master processor is a DP, default device names are *s0* through *s5*.

Disks connected to a processor other than the master processor must be identified in the operating system configuration file. See the *:MassStorage:* parameter in Section 16, "Configuring Operating Systems," for information about assigning device names.

Figure 11-2. Disks in an SRP Primary Cabinet (Rear View)



## Initializing an Unformatted Disk

A new disk that has never been used on a CTOS system must be formatted and initialized before it is recognized as a valid volume. For some disks, that involves a simple procedure, because device-specific parameters and bad spots are already stored on the disk. For other disks, however, you must supply the device template and the bad spot information to the **Format Disk** command.

Table 11-1 lists the device template and bad spot requirements for different models of disks. If a device template or a bad spots file is required, you will need to supply that information in the *[Device template]* or *[Bad spot file]* fields of the **Format Disk** command form. See “Creating a Bad Spots File” and “Using Device Templates,” later in this section, for more detailed information.

**Table 11-1. Device Template and Bad Spot Requirements for Disks**

<b>Disk Model</b>	<b>Device Template Required?</b>	<b>Bad Spots File Required?</b>
SCSI (any)	No	No*
IDE (EISA/ISA workstation)	No	No
Non-SCSI workstation module	Yes	Yes
Non-SCSI ST-506 (SRP)	Yes	Yes
Non-SCSI SMD (SRP)	Yes	Yes
Valid volume (any)	No	No

\*A bad spots file is required only to add bad spots that have occurred since the disk was last initialized. See "Creating a Bad Spots File," later in this section.

The following procedure describes how to format and initialize a disk. This procedure works for most workstation disks; it does not, however, initialize a system volume and it may not initialize the disk optimally for your purposes. For more detailed information, see "Using Parameter Templates," later in this section.

For information about reinitializing a valid volume, see "Reinitializing a Valid Volume," later in this section. If a disk has been corrupted, see "Reinitializing Corrupted Volumes," later in this section.

**Caution**

The following procedure destroys all data on the disk.

1. If you are formatting an SRP disk, start a Cluster View session on the processor to which the disk is connected.
2. On the Executive command line, type **Format Disk**; then press **RETURN**.



3. Fill in the command form as shown in the following example; parameter fields are described in Table 11-2.

Format Disk

Device name	d1
[Device password]	##
[Current volume password]	
[New volume name]	Volume1
[New volume password]	####
[Configuration file]	
[Format template]	
[Device template]	(see Table 11-1)
[Print file]	
[Overwrite ok?]	
[Bad spot file]	(see Table 11-1)
[Recalculate defaults?]	
[CTOS partition size in Mb]	

4. Press GO.

Table 11-2. Format Disk Parameters

Field Name	Description
<i>Device name</i>	Enter the device name of the disk you want to initialize. When formatting floppy disks, be sure to use the correct type of diskette for your drive; see the <i>CTOS Media User's Guide</i> .
<i>[Device password]</i>	Default: None  Enter the device password for the disk you want to initialize. If you are using a prebuilt CTOS operating system, device passwords for hard disks match device names, and floppy drives do not have passwords. If you are using a customized operating system, however, device passwords might be different.
<i>[Current volume password]</i>	Default: Active password  If you are reinitializing a disk, enter the volume password currently assigned to the disk. When you are initializing a new disk, there is no volume password to enter here.

continued

**Table 11-2. Format Disk Parameters (cont.)**

Field Name	Description
<i>[New volume name]</i>	<p>Default: Current volume name</p> <p>Enter a name, up to twelve characters long, to assign to the volume. It can contain letters, numerals, periods, and hyphens. The volume name of each disk within a cluster must be unique.</p>
<i>[New volume password]</i>	<p>Default: None</p> <p>Enter a password, up to twelve characters long, to assign to the volume. It can contain letters, numerals, periods, and hyphens.</p>
<i>[Configuration file]</i>	<p>Default: See below</p> <p>Enter the name of the configuration file containing the format and device templates for the disk. <i>[Sys]&lt;Sys&gt;FormatDiskConfig.sys</i> is the default. (See "Using Parameter Templates," later in this section.)</p>
<i>[Format template]</i>	<p>Default: DiskDefault for hard disks; FloppyDefault for floppy disks</p> <p>Enter the name of a format template for the disk. If you leave this field blank, a default template is used. If a default template is not available, <b>Format Disk</b> calculates default values. For more information, see "Using Format Templates," later in this section.</p>
<i>[Device template]</i>	<p>Default: See below</p> <p>This parameter is not required when reinitializing a valid volume.</p> <p>For SCSI disks and IDE drives, leave this field blank.</p> <p>For unformatted non-SCSI disks (other than IDE), enter the name of the device template for the disk. For workstation disks, the template usually corresponds to the single-character disk type labeled on the bottom on the disk drive module. If the disk type is unknown, enter the actual device characteristics in the form of <i>Cylinders/Heads/Sectors</i>. If you leave this field blank, <b>Format Disk</b> attempts to format the disk with default parameters, which may not be optimal for the disk.</p>

---

continued

Table 11-2. Format Disk Parameters (cont.)

Field Name	Description
<i>[Print file]</i>	<p>Default: None</p> <p>To record command output in a log file, enter a file specification in this field. The file must be created on a valid volume; that is, you cannot write a log file to the same disk you are formatting.</p>
<i>[Overwrite ok?]</i>	<p>Default: Ask for confirmation</p> <p>If you want to reinitialize a valid volume, enter <b>Yes</b>. If you do not want to overwrite a valid volume, enter <b>No</b>. If you leave this field blank, you are prompted to confirm or cancel the initialization procedure if the disk is a valid volume.</p>
<i>[Bad spot file]</i>	<p>Default: None</p> <p>Enter the file specification of a file containing bad spot information for the disk. This file is required if bad spot information is not available on the disk or to add bad spots that have occurred since the disk was last initialized. The bad spots file must be created on a disk other than the one you want to initialize. See Table 11-1 and "Creating a Bad Spots File," in this section.</p>
<i>[Recalculate defaults?]</i>	<p>Default: No</p> <p>This field applies to volumes that are already initialized. Enter <b>Yes</b> to reinitialize the disk with default values calculated internally by the <b>Format Disk</b> command. If you enter <b>No</b> or leave this field blank, the disk is reinitialized with its current parameters. This parameter does not override parameters contained in the specified format template.</p>
<i>[CTOS partition size in Mb]</i>	<p>Default: All available disk space</p> <p>This parameter applies to EISA/ISA-bus workstations only. Enter a size for the CTOS partition, in M bytes. If you do not plan to install MS-DOS on the workstation, leave this field blank. See "Formatting a Partitioned Disk," later in this section, for more information about disk partitioning.</p>

### Creating a Bad Spots File

A bad spots file is required by the **Format Disk** command if bad spot information is not available on the disk; see Table 11-1 for a list of disks that require a bad spots file.

A bad spots file is also required when you reinitialize a valid volume if you need to add bad spots that have occurred since the last time the disk was initialized. See "Correcting Input/Output (I/O) Errors," later in this section.

If the disk requires a bad spots file, you must create it on a disk other than the one you are initializing before you begin the **Format Disk** command. You can use a floppy disk for this purpose.

To create a bad spots file, use the Editor application, as described in the following procedure.

1. On the Executive command line, type **Editor**; then press **RETURN**.
2. Type a file specification for the bad spots file, as shown in the following example:

```
File name(s)  [f0]<Dir>BadSpots.VolName
```

3. Press **GO**.
4. Enter the list of bad spots. (See the sections about non-SCSI and SCSI bad spot formats, below.)
5. When you have finished entering bad spots, press **FINISH**, then **GO**, to save the file.

### Bad Spot Formats for Non-SCSI Disks

Enter bad spots in one of the formats shown below, leaving a space between each bad spot entry. See the documentation for the disk to determine which format to use.

*c/h/sb/bc*

*c/h/#sector*

*c/h/\$sector*

where

<i>c</i>	Is the cylinder number.
<i>h</i>	Is the head number.
<i>sb</i>	Is the starting byte number.
<i>bc</i>	Is the number of bits in error.
<i>#sector</i>	Is the sector number of a 512-byte sector.
<i>\$sector</i>	Is the sector number of a 256-byte sector.

An example bad spots file is shown in Figure 11-3.

**Figure 11-3. Bad Spots File for a Non-SCSI Disk**

---

```

981/0/#1 877/2/#0 877/2/#1 975/4/#15 969/5/#4 757/6/#15
943/1/#5 943/1/#6 415/4/#15 1000/5/#0 24/2/#0 425/4/#9
880/5/#5 1005/5/#8 1005/5/#9 32/2/#8 737/4/#12 924/5/#8
924/5/#9 494/6/#1 849/2/#12 849/2/#13 885/4/#4 941/5/#9
941/5/#10 524/6/#7 995/4/#3 995/4/#4

```

---

## Bad Spot Format for SCSI Disks

A bad spots file for a SCSI disk should contain only those bad spots reported by the **PLog** command since the last time the disk was initialized. Manufacturer's bad spots and those you have previously specified during a volume initialization are already known to the SCSI disk. Specifying them a subsequent time reduces usable disk space.

To enter a bad spot, type its number, as listed in the system error log. For SCSI disks, that is a 2- to 9- digit number, for example, 12339. When entering more than one bad spot, separate each entry with a space.

## Reinitializing a Valid Volume

You might need to reinitialize a valid volume before restoring a backup or reusing the disk for another purpose. When you reinitialize a valid volume, device parameters and bad spots are taken from the disk, so you do not need to specify either a device template or a bad spots file in the command form. In addition, you have the choice of reinitializing the disk using current volume parameters, such as number of files on the volume and size of the <Sys> directory, or specifying new ones.

The following procedure describes how to reinitialize disks on both workstation and SRPs, using current volume parameters. For information about changing volume parameters, see “Using Format Templates,” later in this section.

**Caution**

The following procedure destroys all data on the disk.

1. Start the **Format Disk** command as described earlier in this section.
2. Fill in the command form as shown in the following example (see Table 11-2 for parameter descriptions).

Format Disk

Device name	d2
[Device password]	##
[Current volume password]	###
[New volume name]	Engr2
[New volume password]	sluggo
[Configuration file]	
[Format template]	
[Device template]	
[Print file]	
[Overwrite ok?]	yes
[Bad spot file]	
[Recalculate defaults?]	
[CTOS partition size in Mb]	

3. Press **GO**.



## Using Parameter Templates

To initialize a disk, the **Format Disk** command obtains input from parameter templates. Such templates are contained in a configuration file named *[Sys]<Sys>FormatDiskConfig.sys*. That configuration file contains the following parameter templates:

Format templates	Contain volume parameters, such as the maximum number of directories and files on the volume
Device templates	Contain disk hardware parameters

The configuration file supplied with Standard Software contains a variety of format templates. You can select from one of them, or you can create your own format template, as described in this section. If you do not specify a particular format template, the volume is initialized with default values.

The configuration file also contains device templates, which supply hardware parameters for specific models of disks. The configuration file contains device templates for most workstation and SRP disks; therefore, you rarely need to create new device templates. A procedure for creating device templates, however, is included later in this section.

### Configuration File Format

Each line of the configuration file is written in the following format:

*:Keyword:Value*

where

*:Keyword:* Identifies a parameter; keywords and the placement of colons must not be changed.

*Value* Is a parameter value.

Keyword parameters for both format templates and device templates are described later in this section. Note that the order of format and device templates is not significant. They can be intermixed and can appear in any order within the configuration file.

### Using Format Templates

The format templates supplied with Standard Software are listed in Table 11-3. To use a particular format template, specify its name in the *[Format template]* field of the **Format Disk** command form. A sample format template is shown in Figure 11-4. Instructions for creating or modifying a format template are included below.

**Figure 11-4. Format Template**

---

```
:FormatTemplate:WSSysVolume
:MaxFilesOnVolume:
:PrimaryFileHeadersOnly?:
:MaxDirectories:
:MaxFilesInSysDirectory:
:PasswordEncryption?:
:Debug?:
:SysDirectoryPassword:
:ProtectSysDirectory?:
:SystemImageSize:768
:CrashFileSize:2048
:SystemLogFileSize:48
:SuppressFormat?:No
:SurfaceTestsIfUnformatted:4
:SurfaceTestsIfFormatted:1
:OldCTOSFormat?:No
```

---

**Table 11-3. Format Disk Standard Format Templates**

Template Name	Characteristics
<b>Defaults</b> (if no format template is specified)	
FloppyDefault	Floppy diskette primary file headers only
DiskDefault	System volume without password protection
<b>Floppy Volumes</b>	
FloppyArchive	Floppy diskette suitable for backups
FloppyData	Floppy diskette with approximately 60 file headers
FloppyProtect	Floppy diskette with password protection and a write-protected <Sys> directory
<b>X-Bus+ and EISA/ISA System Volumes</b>	
SGenSysVolume	X-Bus+ or EISA/ISA system volume without password protection
SGenSysVolumeProtect	X-Bus+ or EISA/ISA system volume with password protection and a write-protected <Sys> directory
SGenSysVolumeEncrypt	X-Bus+ or EISA/ISA system volume with encrypted password protection and a write-protected <Sys> directory
SGenSavePartition	X-Bus+ or EISA/ISA system volume with an existing non-CTOS partition that you do not want to destroy
SGenSavePartitionProtect	X-Bus+ or EISA/ISA system volume with password protection, a write-protected <Sys> directory, and an existing non-CTOS partition that you do not want to destroy
SGenSavePartitionEncrypt	X-Bus+ or EISA/ISA system volume with encrypted password protection, a write-protected <Sys> directory, and an existing non-CTOS partition that you do not want to destroy

continued

**Table 3-3. Format Disk Standard Format Templates (cont.)**

---

<b>Template Name</b>	<b>Characteristics</b>
<b>X-Bus Workstation System Volumes</b>	
WSSysVolume	Workstation system volume without password protection
WSSysVolumeProtect	Workstation system volume with password protection and a write-protected <Sys> directory
WSSysVolumeEncrypt	Workstation system volume with encrypted password protection and a write-protected <Sys> directory
<b>SRP System Volumes</b>	
SrpSCSISysVolume	SRP SCSI system volume without password protection
SrpSCSISysVolumeProtect	SRP SCSI system volume with password protection and a write-protected <Sys> directory
SrpSCSISysVolumeEncrypt	SRP SCSI system volume with encrypted password protection and a write-protected <Sys> directory
SrpSMDSysVolume	SRP SMD system volume without password protection
SrpSMDSysVolumeProtect	SRP SMD system volume with password protection and a write-protected <Sys> directory
SrpSMDSysVolumeEncrypt	SRP SMD system volume with encrypted password protection and a write-protected <Sys> directory
SrpWinSysVolume	SRP ST-506 system volume without password protection
SrpWinSysVolumeProtect	SRP ST-506 system volume with password protection and a write-protected <Sys> directory
SrpWinSysVolumeEncrypt	SRP ST-506 system volume with encrypted password protection and a write-protected <Sys> directory

---

continued

**Table 3-3. Format Disk Standard Format Templates (cont.)**

Template Name	Characteristics
<b>Data Storage Volumes</b>	
DataVolume	Data storage only (a non-system volume) without password protection (workstation or SRP)
SmallDataVolume	Data storage only (a non-system volume) without password protection and approximately 500 file headers (workstation or SRP)
SrpDataVolume	Data storage only (a non-system volume) without password protection (SRP only)
<b>Memory Disks</b>	
MemDisk1	1-megabyte SRP memory disk
MemDisk3	3-megabytes SRP memory disk

### Adding a Format Template

If none of the standard format templates is suitable, you can add one of your own. To do so, use the Editor application, as described in the following procedure:

1. On the Executive command line, type **Editor**; then press **RETURN**.
2. Type **[Sys]<Sys>FormatDiskConfig.sys**, as shown below; then press **GO**.

```
[File name(s)]  [Sys]<Sys>FormatDiskConfig.sys
```

(To add a format template for an SRP disk, specify **[!Sys]<Sys>FormatDiskConfig.sys**.)

3. Move the cursor to the end of the file.
4. If necessary, press **RETURN** to move the cursor to a new line.
5. Type the keyword and value for each parameter (parameters are described below).

6. When you have finished editing, press **FINISH**, then **GO**, to exit the Editor and save the file.

See Section 4, "Using Administrative Tools," for more detailed information about using the Editor.

### Format Template Parameters

The following parameters are included in format templates. The *:FormatTemplate:* keyword must be the first parameter in a format template, and you must assign it a unique value. The other keywords may appear in any order. (They are listed alphabetically below.) If a keyword is omitted or its value left blank, a default value from the **Format Disk** program is used. If you specify **Yes** in the *[Recalculate defaults?]* field of the command form, default values are determined as if the disk is unformatted.

If you do not specify a format template, the template named **DiskDefault** in *[Sys]<Sys>FormatDiskConfig.sys* is used. If that format template does not exist, a default value from the **Format Disk** program is used. Cases where default values differ between the **DiskDefault** template and the **Format Disk** command are noted in the parameter descriptions.

#### *:FormatTemplate:*

This must be the first keyword in each format template and a unique value must be specified. Specify a unique name for the format template.

#### *:CrashFileSize:*

Default for valid volume: Current value

Value in **DiskDefault** template: 8192

Default from **Format Disk**: 0

For nonsystem volumes, specify **0**. For CTOS I system volumes, specify **2048**. For CTOS II and CTOS III system volumes, specify the number of sectors required for *[Sys]<Sys>CrashDump.sys*, the crash dump file. To calculate that number, multiply the amount of processor memory by two.



On X-Bus and X-Bus+ workstations only, you can conserve disk space on *[Sys]* by specifying **2048** to receive the first megabyte of the memory dump. You then create an extended crash dump file on another disk to receive the entire memory dump. See “Collecting a Crash Dump,” in Section 19, “Troubleshooting.”

***:Debug?:***

Default: No

Enter **Yes** to display an **F** for each track that is formatted and a **T** for each track that is surface tested. Enter **No** to suppress this information.

***:MaxDirectories:***

Default for valid volume: Current value

Value in DiskDefault template: Blank

Default from **Format Disk**: See below

Specify the maximum number of directories for the volume. If you do not specify a value, **Format Disk** calculates a value based on the size of the disk.

***:MaxFilesInSysDirectory:***

Default for valid volume: Current value

Value in DiskDefault template: Blank

Default from **Format Disk**: See below

Specify the maximum number of files to be stored in the *<Sys>* directory. For system volumes, 1000 to 2500 files are recommended. If you do not specify a value, **Format Disk** calculates a value based on the size of the disk.

***:MaxFilesOnVolume:***

Default for valid volume: Current value

Value in DiskDefault template: Blank

Default from **Format Disk**: See below

Specify the maximum number of files to be stored on the volume. The maximum number of files is based on twenty-five sectors per file; therefore, the number you specify here may be adjusted by **Format Disk** according to the size of the disk.

To specify an exact number, precede this value with a pound sign (#), for example, #5000. When the value is preceded by a pound sign, **Format Disk** attempts to create the exact number of file headers specified; in cases where secondary file headers are requested, however, **Format Disk** creates additional file headers. If the number of files requested seems too large for the size of the disk, **Format Disk** reduces the number of files to a reasonable value.

If you do not specify a value, **Format Disk** calculates a value based on the size of the disk.

### ***:OldCTOSFormat?***

:Default: No

Enter **No** to format the disk such that it describes a logical block address (LBA) device. (For information about LBA devices, see the *CTOS Procedural Interface Reference Manual*). Such a disk is formatted optimally for its device characteristics and is for use with 3.3 CTOS I and II, 1.0 CTOS III, and 3.0 CTOS/XE (and subsequent higher versions) only. Such a disk is not usable on systems running an earlier version of the operating system.

If you enter **Yes**, the disk is formatted to be backward-compatible with earlier versions of the operating system; that is, versions earlier than 3.3 CTOS I or II and 3.0 CTOS/XE. With SCSI disks, however, such backward compatibility may result in disks that are not formatted to full capacity and that run more slowly than is optimal. For floppy disks, it is recommended that you specify **Yes** to this parameter.

### ***:PasswordEncryption?:***

Default: No

Enter **Yes** if you want the password to be encrypted. This provides security against sophisticated users who might be able to “peek” at the volume password. If you encrypt the password, though, keep careful records, because it cannot be deciphered. Enter **No** if you do not want the password to be encrypted.

***:PrimaryFileHeadersOnly?:***

Default: No

Enter **Yes** if you want only primary file headers on the volume. This option is usually used for floppy disks only. If you enter **No** or leave this field blank, secondary file headers are created. These are needed to recover data from a corrupted volume.

***:ProtectSysDirectory?:***

Default for valid volume: Current value

Default for unformatted disk: No

Default: No

Enter **Yes** to set the protection level of <Sys> to 5, which prevents users from changing or adding files without a password. (See "Protecting Directories," in Section 6, "Implementing System Security," for more information about protection levels.) Enter **No** to set <Sys> directory protection to 15, which does not require a password to change or create files.

***:SuppressDefaultScsiPages?:***

Default: No

Enter **Yes** to use the current SCSI page settings. If you enter **No** or leave this field blank, default SCSI page parameters are used.

***:SuppressFormat?:***

Default: No

Enter **No** if the disk has never been formatted or if diagnostics have been run on it. If you are reinitializing a valid volume, enter **Yes**; this decreases the time required to initialize a disk. If you enter **Yes** and the disk is not a valid volume, the **Format Disk** command fails with an input/output (I/O) error (301).

***:SuppressVolumeStructures?:***

Default: No

Enter **Yes** if you do not want CTOS volume structures to be created on this disk. This reserves the entire disk for future use as a non-CTOS partition.

### ***:SurfaceTestsIfFormatted:***

Default: 1

Specify the number of surface tests to be run on a formatted disk. At least one surface test is recommended.

### ***:SurfaceTestsIfUnformatted:***

Default: 4

Specify the number of surface tests to run on an unformatted disk (a new disk or after disk diagnostics). It is not unusual for a disk to have bad spots; however, it is important that they be detected so that no data is stored on them.

*Note:* Run surfaces tests in multiples of 4 to ensure that all data-storage patterns on the disk are tested.

### ***:SysDirectoryPassword:***

Default for valid volume: Current value

Default for unformatted disk: None

Specify a password, up to twelve characters long, to assign to the <Sys> directory.

### ***:SystemImageSize:***

Default for valid volume: Current value

Value in DiskDefault template: 1280

Default from **Format Disk**: 0

For nonsystem volumes, specify 0. For system volumes, specify the number of sectors for *[Sys]<Sys>SysImage.sys*, the operating system file. The size of this file cannot be changed after you have initialized the disk. For prebuilt operating systems, 1280 is the recommended value. If you are using a customized operating system, you may need a larger value.

**:SystemLogFileSize:**

Default for valid volume: Current value  
Value in DiskDefault template: 48  
Default from **Format Disk**: 0

For nonsystem volumes, specify **0**. For system volumes, enter a number of sectors for *[Sys]<Sys>Log.sys*, the system log file. To track system problems accurately, 48 sectors is recommended. If your system volume is small, however, you can specify a smaller value (for example, 10 or 20 sectors).

**:Verify?:**

Default: No

Enter **Yes** to verify that the disk has been formatted and initialized correctly. Enter **No** to bypass the verification operation.

## Using Device Templates

The device templates supplied with Standard Software are listed in Table 11-4. To use a particular device template, specify its name in the *[Device template]* field of the **Format Disk** command form. A sample device template is shown in Figure 11-5, and instructions for creating a device template are included later in this section.

For SCSI disks and IDE drives, do not specify a device template; **Format Disk** obtains device parameters directly from the hardware. When formatting another type of disks or a corrupted volume, however, you may need to specify a device template. See Table 11-1, earlier in this section.

If a device template is required and you do not specify one, or if the one you specify does not exist, **Format Disk** uses the device template named "Unknown." This device template might not contain optimal parameters for the disk, or it might cause the **Format Disk** command to fail. In such a case, you must add a device template to the configuration file for the **Format Disk** command.

**Figure 11-5. Device Template**

---

```
:DeviceTemplate:Fujitsu80
:CylindersPerDisk:589
:TracksPerCylinder:7
:SectorsPerTrack:32
:BytesPerSector:512
:Removable?:No
:SeekStepRate:14
```

---

### Adding a Device Template

You add a device template to *FormatDiskConfig.sys* in the same way you add a format template. See “Adding a Format Template,” earlier in this section.

**Table 11-4. Format Disk Standard Device Templates**

---

<b>Manufacturer and Size of Disk</b>	<b>Device Template Name</b>
Workstation disk vendor codes	A to Z and A1 to Z1
Atasi, 46 megabytes	Atasi46
Ball, 100 megabytes	Ball100
Control Data, 300 megabytes	CDC300
Control Data, 340 megabytes	CDC340
Control Data, 675 megabytes	CDC675
Fujitsu, 80 megabytes	Fujitsu80
Hitachi, 51 megabytes	Hitachi51
Hitachi, 85 megabytes	Hitachi85

---

continued



Table 11-4. Format Disk Standard Device Templates (cont.)

Manufacturer and Size of Disk	Device Template Name
Maxtor, 53 megabytes	Maxtor53
Maxtor, 143 megabytes	Maxtor143
Memorex, 166 megabytes	Memorex166
Micropolis, 52 megabytes	Micropolis52
Micropolis, 85 megabytes	Micropolis85
Nortel, 350 megabytes	Nortel350
Toshiba, 85 megabytes	Toshiba85
Memory disk, 1 megabyte	MemDisk1
Memory disk, 3 megabytes	MemDisk3
Regular capacity floppy disk	FloppyType
High-capacity floppy disk	FloppyTypeHiCap

### Device Template Parameters

Keywords and parameter values for device templates are listed below. They are included here in case you need to create a new device template. *Do not change parameters in the device templates supplied with Standard Software.* To determine parameter values for new device templates, see the documentation for the disk. The `:DeviceTemplate:` keyword must be the first parameter in a device template, and you must assign it a unique value. The other keywords can appear in any order; they are listed alphabetically below.

#### ***:DeviceTemplate:***

This must be the first keyword in each device template. Enter a unique name for each device template.

***:BytesPerSector:***

Default: 512

Enter the number of bytes to be allocated in each sector.

***:CylindersPerDisk:***

Default: 306

Enter the number of cylinders to be allocated on the disk.

***:HiCapacityFloppy?:***

Default: No

Enter **Yes** to format high-capacity floppy disks on a high-capacity drive. Enter **No** to format regular-capacity floppy disks on a high-capacity drive. Be sure to use the correct type of floppy disk for your drive; see the *CTOS Media User's Guide*. This field has no effect on regular-capacity floppy drives.

***:Removable?:***

Default: No

Enter **Yes** if the disk is a removable storage medium.

***:SectorsPerTrack:***

Default: 16

Enter the number of sectors to be formatted for each track. On some SRP disks, you can increase usable disk space by changing this number to 17. This is the only parameter that should be changed in any of the device templates supplied with Standard Software.

***:SeekStepRate:***

Default: 0

Enter a number to represent the time interval, in microseconds, between successive step pulses when a seek command is issued. (See the Western Digital WD-2010 documentation.) Common values are 0 (the default) and 14.

***:SpiralFactor:***

Default: 0

This field applies to SMD disks only; it has no effect on other disks. Specify the sector offset from track to track on the disk.

***:TracksPerCylinder:***

Default: 4

Enter the number of tracks to be formatted for each cylinder.

***:UtilizeEcc?:***

Default: See below

Enter **Yes** to use error checking and correction (ECC) format.

The default is set according to the capabilities of the hardware. If you specify **Yes**, but ECC capability is not present, the disk is formatted in cyclic redundancy check (CRC) mode. Specify **No** if ECC capability is present but you do not want to use it.

***:WritePreCompCyl:***

Default: 0

Enter the number of the cylinder at which write-precompensation begins. Write-precompensation reduces I/O errors on inner cylinders of the disk. The default, 0, specifies no write-precompensation and is used on disks that do not provide this feature.

## Using the Configure Disk Command

The **Format Disk** command provides the advantage of reusable format templates so that you can save and keep track of the parameters you use to initialize a disk. It is possible, however, to initialize a disk without using a format template. The **Configure Disk** command, also supplied with Standard Software, contains the volume parameter fields in its command form, as shown below:

```
Configure Disk
Device name _____
[Device password] _____
[Current volume password] _____
New volume name _____
[New volume password] _____
[System image size] _____
[Log file size] _____
[Crash file size] _____
[Max. directories] _____
[Max. files on volume] _____
[Primary file headers only?] _____
[Max. files in <Sys>] _____
[Suppress format of medium?] _____
[Surface tests] _____
[Debug=D, password encrypt=P] _____
[Print file] _____
[Bad Spot file] _____
[Old CTOS format?] _____
[Format template] _____
[Device template] _____
[CTOS Partition in Mb] _____
```

Parameters for the **Configure Disk** command are a combination of the **Format Disk** fields, described in Table 11-2, and the format template parameters described in this section. See the *CTOS Executive Reference Manual* for more detailed information about the **Configure Disk** command.

## Formatting a Partitioned Disk

On EISA/ISA workstations, you can format a disk so that it contains more than one partition. One partition can be used for CTOS files and another for non-CTOS files, such as those associated with DOS and its applications.

To format multiple partitions with the **Format Disk** (or **Configure Disk**) command, you specify a size for the CTOS partition, as shown in the following example. That reserves the remaining disk space to be used as a non-CTOS partition.

```

Format Disk
Device name                d0
[Device password]         ##
[Current volume password] #####
[New volume name]         CTOSPar
[New volume password]     #####
[Configuration file]
[Format template]         SgenSysVolume
[Device template]
[Print file]
[Overwrite ok?]           yes
[Bad spot file]
[Recalculate defaults?]
[CTOS partition size in Mb] 80

```

## Activating a Bootable non-CTOS Partition

When a disk contains more than one bootable partition, only one of those partitions can be active on the workstation at any given time. If you are using CTOS and want to change the active partition on a disk, you use the **Disk Partition Set Active** command, as described in the following steps:

1. On the Executive command line, type **Disk Partition Set Active**; then press **RETURN**.

2. Fill in the command form, as shown in the following example:

Disk Partition Set Active	
Device name	d0
[Device or volume password]	####
Partition ID	3

3. Press **GO** to activate the partition.
4. Press the Reset button or On/Off switch to boot from the non-CTOS partition. (The non-CTOS partition must be located on *d0*.)

In most cases, the ID for the non-CTOS partition is 3; however, you can use the **Disk Partition Display Details** command to display a list of partition IDs. See the *CTOS Executive Reference Manual* for information about using that command.

## Reactivating a CTOS Partition

The procedure for reactivating a CTOS partition depends on the software that is running on the workstation. With DOS, for example, you use the **FDISK** command to specify the appropriate non-DOS partition (usually partition 2) as the active partition. You then reboot the workstation to boot from the CTOS partition.

See the documentation for your other-than-CTOS software for more specific information.

## Correcting Input/Output (I/O) Errors

When you receive a disk from the manufacturer, it has been surface-tested and known bad spots are accounted for. Known bad spots are normal and do not cause loss of data if they are properly specified when the disk is initialized.

As you use the disk, however, additional bad spots may occur; these are a serious problem and can cause loss of data. New bad spots are signaled by I/O errors (Error 301) when you try to create or modify files. When I/O errors occur, you need to make bad spots known to the disk so that they will not cause loss of data.



As bad spots occur, they are reported in the system error log, which you can view or print with the **PLog** command. To make bad spots known to a disk, you enter them into a bad spots file and then reinitialize the disk, using the bad spots file as a parameter value. Procedures for creating a bad spots file and reinitializing the disk are described below.

*Note: If new bad spots occur frequently, back up the disk and replace it with a new one as soon as possible.*

## Specifying Bad Spots

As bad spots occur, they are reported in the system error log, which you can view or print with the **PLog** command. You need to make those bad spots known to the disk so that data will not be stored on them. Bad spots can be defined when a disk is reinitialized, or on non-SCSI disks, with the **Bad Sector** command.

To define bad spots when you reinitialize a disk, you enter them into a text file that you create with the Editor application. You then specify the name of that file in the *[Bad spot file]* field of the **Format Disk** command form. See "Creating a Bad Spots File," earlier in this section, for a step-by-step procedure.

For information about using the **Bad Sector** command on non-SCSI disks (including IDE drives), see the documentation for it in the *CTOS Executive Reference Manual*.

### Caution

A bad spots file for a SCSI disk should contain only those bad spots reported by **PLog** since the last time the disk was initialized. Respecifying bad spots that are already known to the disk reduces the amount of usable disk space.

## Running Surface Tests

After creating a bad spots file, you must reinitialize the disk to make the bad spots known to the disk. When you do so, run at least four surface tests to detect additional bad spots. (It is important to run surface tests in multiples of 4 to make sure that the entire disk gets tested.)

## Adding Hard Disks

---

To run more than one surface test (which is the default), modify the *:SurfaceTestsIfFormatted:* parameter in the appropriate format template. The following example shows you how to specify four surface tests:

*:SurfaceTestsIfFormatted:4*

## Reinitializing the Disk

### Caution

The following procedure destroys all data on the disk.

Before you reinitialize the disk, back it up so that you can restore directories and files to it later. See Section 13, "Backing Up and Restoring Data."

The following example shows you how to specify a format template and a bad spots file in the **Format Disk** command form:

Format Disk	
Device name	d1
[Device password]	##
[Current volume password]	###
[New volume name]	Volume1
[New volume password]	###
[Configuration file]	
[Format template]	DataVolume
[Device template]	
[Print file]	
[Overwrite ok?]	
[Bad spot file]	[d1]<Sys>BdSpt.d1
[Recalculate defaults?]	
[CTOS partition size in Mb]	

For more detailed information, see the step-by-step procedures provided earlier in this section.

## Reinitializing Corrupted Volumes

A *corrupted volume* is not recognized by the operating system. The following conditions can cause a corrupted volume:

- The volume home block (VHB) becomes unreadable, usually because of bad spots on the disk.
- The **Format Disk** command is abnormally terminated, possibly because of equipment failure, or because you pressed **ACTION+FINISH**.
- Diagnostics have been run on the disk. (See the documentation for your diagnostics software product.)

In many cases, you can recover data from a corrupted volume with the **Volume Archive** command.

To reinitialize a corrupted volume, see the procedures earlier in this section for entering a device template and bad spots file in the **Format Disk** command form.

## Optimizing Disk Space

In some cases, you can increase available disk space by adjusting format template parameters. When you are just beginning to use a system, it is difficult to know exactly what your needs will be. Therefore, the default parameters supplied in `[Sys]<Sys>FormatDiskConfig.sys` are a good place to start. They are based on averages of typical file lengths and number of files per directory.

As you use a disk, however, a discrepancy between available disk space (sectors) and file headers shows that the disk is not initialized optimally. To determine this, use the **Volume Status** command, as described below:

1. On the Executive command line, type **Volume Status**; then press **RETURN**.
2. Type the volume or device name of the disk you want to check; then press **GO**.

An example of the top portion of the **Volume Status** display is shown in Figure 11-6. The column labeled "Used" (far right) shows the percentage of sectors and file headers currently used on the disk. Sectors refer to disk storage space; file headers refer to slots for file names.

When a disk is initialized optimally, the percentage of used sectors and file headers should be similar. As shown in Figure 11-6, many more sectors than file headers have been used. You can optimize disk space by reinitializing the volume.

**Figure 11-6. Volume Status Display**

---

Status of volume TFS-1	Device d1			
Initialized	Apr 21, 1993	3:14 PM		
Last Modified	Aug 30, 1993	1:16 PM		
	Unused	Reserved	CTOS Volume Total	Used
Sectors	38138	0	131072	70%
File headers	2519	0	3719	32%

---

**Caution**

Reinitializing a volume destroys all data on it. Therefore, be sure to back up the volume before you reinitialize it. See Section 13, "Backing Up and Restoring Data," for backup and restoration procedures.

The volume shown in Figure 11-6 was initialized with too many file headers. Because file headers occupy disk space, they are taking up space that could be used for other data. The disk will be full of data long before all the file headers are used. Therefore, to optimize disk space, you could reinitialize the disk with fewer file headers. To do so, adjust the value for the *:MaxFilesOnVolume:* parameter in the format template.

The opposite situation occurs when you do not allocate enough file headers. In that case, you can run out of file headers while free sectors remain on the disk. This results in the following error when you try to create a file:

```
No free file headers (Error 225)
```

Another common misuse of disk space is allocating a large number of file headers in the <Sys> directory of a nonsystem volume. On nonsystem volumes, <Sys> usually contains only the mandatory files that are created when the disk is initialized. Therefore, you can specify a small number of files, for example, 25, in <Sys>, to make space available for other directories. To do that, adjust the value for the *:MaxFilesInSysDirectory:* parameter in the format template.





# Section 12

## Using Tape and CD-ROM Drives

### Tape Drives

Your workstation or SRP may be equipped with one of the following types of tape drives:

#### **SCSI quarter-inch cartridge (QIC)**

SCSI QIC tape drives can be found on both workstation and SRPs. They use QIC tape cartridges and write up to 150M bytes of data per 600-foot tape. A SCSI tape drive can read QIC tapes created on either SCSI or non-SCSI tape drives.

#### **Non-SCSI QIC**

Non-SCSI QIC tape drives can also be found on both workstations and SRPs. They use QIC tape cartridges and write up to 60M bytes of data per 600-foot tape. A non-SCSI tape drive can read QIC tapes created on non-SCSI tape drives only.

#### **Digital data storage (DDS)**

DDS drives are available as workstation modules, Series 5000 cartridges, or as built-in units in certain models of EISA/ISA workstations. They use DDS cartridge tapes and write up to 1.3G bytes (1300M bytes) of data per 60-meter tape.

#### **Half-inch**

Half-inch tape drives are available on SRPs only. They are connected to a Storage Processor and use reel-to-reel tapes. The amount of data that can be written to a half-inch tape varies according to tape length and recording density.

See the *CTOS Media User's Guide* for information about inserting tapes, write-protecting tapes, and tape storage specifications.

### What Kind of Tapes to Use

Approved tapes for QIC and DDS tape drives are listed in Table 12-1. For half-inch tape drives, see the owner's manual for information about approved tapes.

**Table 12-1. Approved Tapes for Data Storage**

Drive Type	Tape Model Number	Supplier
Non-SCSI QIC	DC-600A	3M
	DC-615A	3M
	10000FTPI	Data Electronics
SCSI QIC	HD 600XTD	Unisys
DDS	DG-60M	SONY
Half-inch	See the owner's manual for your tape drive	

### Hardware and Software Requirements

Before you can use a tape drive, you must install the Sequential Access Service on the workstation or SRP to which the tape drive is connected. You can do so during system initialization or with an Executive command.

For non-SCSI workstation QIC drives, you also need to install the XBIF Service before you install the Sequential Access Service. See "Where to Install System Services," in Section 9, "Installing System Services."

On SRPs, you must identify tape drives in the operating system configuration file before you install the Sequential Access Service. See the *:SequentialStorage:* parameter in Section 16, "Configuring Operating Systems." Then, install the Sequential Access Service according to the following guidelines:

- For non-SCSI QIC drives, install the Sequential Access Service on any processor except a DP or SP.
- For half-inch tape drives, install the Sequential Access Service on the DP or SP to which the tape drive is connected.

- For SCSI QIC drives, install the Sequential Access Service on any processor, although performance is best when it is installed on the GP to which the tape drive is connected.

In most cases, installing it on the processor to which the tape drive is connected yields the best performance.

### Installing the Sequential Access Service

Before you install the Sequential Access Service, exit any Context Manager or Remote User Manager sessions and, if necessary, start an Administrator Cluster View session. Then, follow these steps:

1. On the Executive command line, type **Install Sequential Access Service**; then press **RETURN** to display the command form:

```
Install Sequential Access Service
```

```
[Device(s) ([QIC])] _____
```

```
[Buffer pool size in Kb (64)] _____
```

```
[QIC interface slot - SRP/XE only (77)] _____
```

2. If necessary, fill in the command form. Parameter fields are described in Table 12-2.
3. Press **GO**.

Alternatively, to install the Sequential Access Service during system initialization, add the following entry to the system initialization JCL file:

```
Run [Sys]<Sys>InstallSeqService.run, Device, BuffSize, Slot
```

Note that you can specify multiple tape drives by using the following syntax with the Device parameter:

```
(Device1, Device2, ..., DeviceN)
```

See "JCL Syntax," in Section 9, "Installing System Services," for more detailed information.

**Table 12-2. Install Sequential Access Service Parameters**

Field Name	Description
<i>[Device(s) ([QIC])]</i>	<p><b>Workstations.</b> Assign a device name for each tape drive. Defaults are <i>[QIC]</i> for the first tape drive and <i>[Seq0]</i> for the second tape drive.</p> <p><b>SRPs.</b> Enter the device name assigned to each tape drive in the operating system configuration file. There is no default name. See the <i>:SequentialStorage:</i> parameter in Section 16, "Configuring Operating Systems."</p>
<i>[Buffer pool size in Kb (64)]</i>	<p>Default: 64</p> <p>Enter the number of K bytes for the buffer pool for the Sequential Access Service. This buffer pool is used by all tape drives on the system. On SRPs, the buffer pool consumes memory on the processor on which you install the Sequential Access Service.</p>
<i>[QIC interface slot - SRP/XE only (77)]</i>	<p>Default: 77h</p> <p>This parameter applies to SRPs only. Enter the slot number, as a hexadecimal number, to which the QIC tape drive is connected.</p>

### Configuring a Tape Drive

By default, the Sequential Access Service configures tape drives to provide acceptable performance in a variety of settings. For optimal performance, however, you may need to modify the configuration file for your tape drive. See the **Configure Sequential Access Device** command in the *CTOS Executive Reference Manual*.

### Preparing Tapes for Use

Before you use a new QIC tape, it is a good idea to retension it. If a tape is not correctly tensioned, it can break or malfunction in the tape drive. To retension a QIC tape, follow these steps:

1. Insert the tape cartridge into the tape drive (see the *CTOS Media User's Guide*).
2. On the Executive command line, type **Tape Retension**.
3. Enter the device name of the tape drive.
4. Press **GO**.

If a tape already contains data, you can erase it before you reuse it. Erasing a tape also retensions it. Note, however, that it can take up to three hours to erase a DDS cartridge. To erase a tape, follow these steps:

1. Insert the tape into the tape drive (see the *CTOS Media User's Guide* or the owner's manual for your tape drive).
2. On the Executive command line, type **Tape Erase**; then press **GO**.

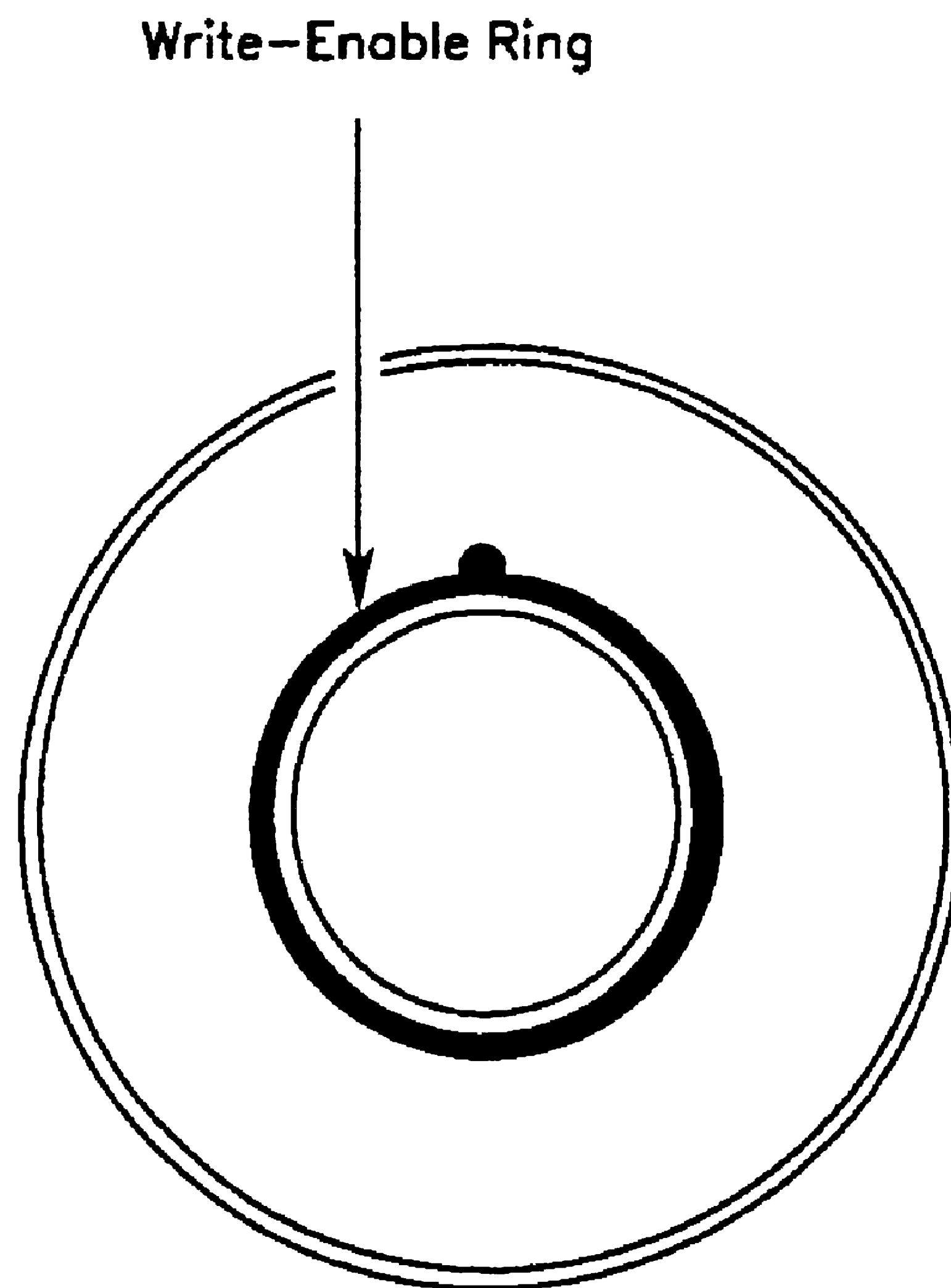
### Write-Enabling Half-Inch Tapes

To write-enable a half-inch tape, place a write-enable ring into the groove on the backside of the tape reel, as shown in Figure 12-1. When the ring is in place, the tape is write-enabled; when it is removed, the tape is write-protected.

See the *CTOS Media User's Guide* for information about write-enabling and write-protecting QIC and DDS tapes.



Figure 12-1. Write-Enable Ring on a Half-Inch Tape



603.12-1

## CD-ROM Drives

Some workstations are equipped with a CD-ROM drive. CD-ROM drives for X-Bus workstations are housed in workstation modules. On EISA/ISA workstations, CD-ROM drives are contained within the system. X-Bus+ workstations can accommodate either a workstation CD-ROM module or an external SCSI CD-ROM drive.

### Hardware and Software Requirements

On multiprocessor systems, the CD-ROM drive functions best when it is connected to the master processor.

Some compact discs (CDs), such as the CTOS software installation CD, are initialized as CTOS volumes. When placed in a CD-ROM drive, such a disc can be accessed like any other disk media. For example, to execute the **Volume Status** command, you simply specify the volume name of the CD or the device name of the CD-ROM drive.



CD-ROM drives bear SCSI device names, *zn*, where *n* specifies the unit number of the SCSI device, as recognized by the operating system.

*Note:* The device naming convention of *zn* applies to all SCSI devices except disks, which follow the disk naming convention of *dn*.

Some CDs, however, such as those that contain the CTOS online documentation library, are created with DOS disc architecture. To access data on that type of CD, you need to install the CD-ROM Service, as described in the following section.

## Installing the CD-ROM Service

Before you install the CD-ROM Service, exit any Context Manager or Remote User Manager sessions and, if necessary, start an Administrator Cluster View session. Then, follow these steps:

1. On the Executive command line, type **Install CDRROM Service**; then press **RETURN** to display the command form:

```
Install CDRROM Service
  [CDROM device names] _____
  [Maximum number of CRDROM files open (300)] _____
  [Maximum number of CDRROM users (100)] _____
  [CDROM volume name (CDROM)] _____
```

2. If necessary, fill in the command form; it is most typical to use default values. Parameter values are described in Table 12-3.
3. Press **GO**.

Alternatively, to install the CD-ROM Service during system initialization, add the following entry to the system initialization JCL file:

**Run [Sys]<Sys>CDROMService.run**

**Table 12-3. Install CD-ROM Service Parameter Fields**

<b>Field Name</b>	<b>Description</b>
<i>[CDROM device names]</i>	<p>Default: See below</p> <p>Enter the device name of the CD-ROM drive that you want recognized by the CD-ROM Service. By default, the CD-ROM Service recognizes devices named <i>zn</i>, where <i>n</i> is the sequential unit number of the nondisk SCSI device (for example, <i>z0</i>, <i>z1</i>, and so on).</p>
<i>[Maximum number of CDROM files open (300)]</i>	<p>Default: 300</p> <p>Specify the total number of CD-ROM files that can be open simultaneously by all users on all CD-ROM drives. For most purposes, the default value of 300 is sufficient. The maximum value is 4095.</p>
<i>[Maximum number of CDROM users (100)]</i>	<p>Default: 100</p> <p>Specify the total number of users who can access CD-ROM files simultaneously. For most purposes, the default value of 100 is sufficient. The maximum value is 65,535.</p>
<i>[CDROM volume name (CDROM)]</i>	<p>Default: CDROM</p> <p>Enter a volume name for the CD-ROM drive.</p>

# Section 13

## Backing Up and Restoring Data

### Performing Routine Backups

Backups protect the data on your system against accidental loss or damage. During a backup, files are copied to an *archive dataset* on tapes or floppy disks. From the archive dataset, you can restore an entire volume or a single file. As a system administrator, you are most likely responsible for performing or supervising backups of the system.

This section describes how to back up data to tape. For information about backing up to floppy disks, see the *CTOS Executive User's Guide*.

The importance of regular backups cannot be overemphasized. After you have become familiar with the procedures, they take only minutes of your time and could save hours of work in trying to recreate the files on your system.

Although you can set up any backup schedule that is convenient for you, the following method is used by many system administrators.

1. Once a week, perform a complete backup of each disk on the system.
2. On the days between complete backups, perform an incremental backup of each disk. An incremental backup archives only those files that were created or changed since a particular date or time.

Procedures for using this method are included later in this section.

### Cleaning Up Disks Before Backups

Before you begin a backup, delete files you no longer need. For example, some volumes contain a "dollar-sign" directory named `<$000>`, which contains temporary files created by applications. Delete files in that directory before you perform a backup. Doing so increases disk space and shortens backup time.

In addition, the following duplicate and temporary files are created by other applications. These files can be deleted periodically, at your discretion.

<i>-old</i> files	File names ending with <i>-old</i> are created by word-processing applications. They contain previous versions of files that have been edited.
<i>.ts</i> files	File names ending with <i>.ts</i> are typescript files produced by text-processing applications. They are required to recover a document after a system crash, power failure, or abnormal termination of the application.
<i>.lst</i> files	File names ending with <i>.lst</i> contain error listings created by compilers or assemblers.

## Performing a Complete Volume Backup

This section describes how to back up data to tapes. Before you begin, make sure that the Sequential Access Service is properly installed and that you have enough tapes to complete the backup. See Section 12, "Using Tape Drives," for more information.

In addition, if you are backing up a disk on the server, you may want to disable the cluster so that users cannot make changes to the disk. To do so, use the **Disable Cluster** command, as described in the *CTOS Executive Reference Manual*.

To perform a complete volume backup, follow these steps:

1. Insert a tape into the tape drive. (See the installation guide for your tape drive.)
2. Retension or erase the tape. (See "Preparing Tapes for Use," in Section 12, "Using Tape and CD-ROM Drives.")
3. On the Executive command line, type **Volume Archive**; then press **RETURN**.

4. Fill in the command form as shown in the following example. Parameter fields are described in Table 13-1. (See also the **Volume Archive** command in the *CTOS Executive Reference Manual* for more detailed parameter descriptions.)

Volume Archive

Volume or device name	d0
[Volume or device password]	####
[Incremental from]	
[Suppress backup?]	
[Suppress verification?]	
[Archive dataset ([QIC])]	[QIC]
[Delete existing archive dataset?]	
[Print file]	
[Display structures?]	
[Verify write?]	
[Suppress user interaction?]	

5. Press **GO** to begin the backup.

After each tape is complete, label it with the following information:

- Type of backup (for example, “full backup” or “complete backup”)
- Volume or device name
- Date of backup
- Sequential tape number (for example, if more than one tape is required to back up a disk, label them 1 of 2, 2 of 2, and so on)



**Table 13-1. Volume Archive Parameters**

Field Name	Description
<i>Volume or device name</i>	Enter the volume or device name of the disk you want to back up.
<i>[Volume or device password]</i>	Default: Active password To back up a valid volume (as is the case for routine backups), enter the volume password. To back up a corrupted volume, enter the device password. (See also "Backing Up a Corrupted Volume," later in this section.)
<i>[Incremental from]</i>	Default: See below To back up files created or modified since a particular date, enter a date, or a date and time, as shown in the following example:  2/4/90 8:30 AM If you leave this field blank, all files on the disk are backed up. If you enter a date only, midnight is the default time.
<i>[Suppress backup?]</i>	Default: No If you leave this field blank, a backup takes place when you execute <b>Volume Archive</b> . However, you can also use the command to verify the integrity of your file system. To run a file system verification only, enter <b>Yes</b> .
<i>[Suppress verification?]</i>	Default: No If you leave this field blank, the file system is verified after a backup. To suppress file system verification, which normally occurs after a backup, enter <b>Yes</b> . (This saves time if you are planning to reinitialize the disk immediately; however, it is not recommended for routine daily backups.)
<i>[Archive dataset (QIC)]</i>	Default: <i>[QIC]</i> Enter the name of the tape device you want to use. For example, <i>[QIC]</i> for a workstation tape drive or, for an SRP tape drive, the name assigned to it in the operating system configuration file.

---

continued



**Table 13-1. Volume Archive Parameters (cont.)**

Field Name	Description
<i>[Delete existing archive dataset?]</i>	<p>Default: Prompt user</p> <p>If you leave this field blank, you are informed if the tape already contains a backup. To overwrite the tape, enter <b>Yes</b>. To prevent overwriting, enter <b>No</b>.</p>
<i>[Print file]</i>	<p>Default: Screen only</p> <p>If you leave this field blank, command output is written to the screen only. To write command output to a file or printer, in addition to the screen, enter a printer or file specification.</p>
<i>[Display structures?]</i>	<p>Default: No</p> <p>If you enter <b>Yes</b>, a detailed analysis of the volume control structures is displayed. This option is generally used only by programmers for file system error analysis.</p>
<i>[Verify write?]</i>	<p>Default: No</p> <p>This field applies only when you archive data to disk backup media, such as floppy disks. It does not apply to tape backups. Enter <b>Yes</b> if you want to verify that the data written to the archive file matches the data you are backing up. If data does not match, an I/O error (301) is reported.</p>
<i>[Suppress user interaction?]</i>	<p>Default: No</p> <p>If you enter <b>Yes</b>, the command exits with an error when user interaction is required. This parameter is most frequently used in software installation scripts. If you leave this field blank, <b>Volume Archive</b> pauses when user interaction is required.</p>

# Performing an Incremental Backup

Many system administrators perform a complete volume backup once a week and an incremental backup on the days in between. This saves time by archiving only those files that were created or changed since the date you specify.

You can use either of the following incremental backup methods. Procedures for restoring data are included later in this section.

### Method 1

Specify the date of the last complete volume backup as the incremental date. This allows you to reuse the same tape day after day, for incremental backups. Each subsequent day, however, more and more files are backed up, so it takes longer to perform the backup.

Should you need to restore data, restore the complete volume backup first, then the incremental backup.

### Method 2

Specify the current date as the incremental date. This decreases backup time, because only a single day's-worth of files are backed up. However, you need more tapes, because each incremental backup must be performed on a separate tape.

Should you need to restore data, restore the complete volume backup first, then each incremental backup in chronological order, starting with the oldest.

To perform an incremental backup, follow the procedure described below for each disk on the system.

1. Insert a prepared tape into the tape drive. (See “Preparing Tapes for Use,” in Section 12, “Using Tape and CD-ROM Drives.”) Do not use the same tape you used for the complete volume backup.
2. Start the **Volume Archive** command with the Executive, as described in the preceding procedure.
3. Enter the incremental date you want to use (see “Method 1” and “Method 2,” above), as shown in the following example:

```

Volume Archive
Volume or device name          d0
[Volume or device password]    ####
[Incremental from]             7/27/93
[Suppress backup?]
[Suppress verification?]
[Archive dataset ([QIC])]      [QIC]
[Delete existing archive file?]
[Print file]
[Display structures?]
[Verify write?]
[Suppress user interaction?]
    
```

4. Press **GO** to begin the backup.

Label the tapes as described for complete volume backups, earlier in this section.

## Performing Backups With Cluster View

You can perform a backup of a disk on a videoless server from any workstation in the cluster. However, you can perform such a backup more quickly if you execute it via Cluster View. If you do not use Cluster View, data travels to the client workstation for processing before the backup is written to tape. With Cluster View, however, data is processed on the videoless workstation or SRP.

To perform backups with Cluster View, you must use a tape drive located on the server. Remember that you do not have access to local disks or tapes when using Cluster View.

To perform a backup with Cluster View, follow these steps:

1. Start a Cluster View session on the SRP. (See “Cluster View,” in Section 4, “Using Administrative Tools.”)
2. Follow the procedures for performing complete volume backups and incremental backups, earlier in this section.

## Restoring Backups

You can restore all or part of an archive dataset. In many cases, restoring data is a routine procedure that does not involve damage to a disk or the file system. The most common routine situations are listed below, and procedures for dealing with them are included in this section.

- You want to transfer all data to a different (perhaps larger) disk; see “Restoring a Complete Backup,” below.
- You receive error messages about a disk being *fragmented*. This happens after a disk has been used for a while. It means that small blank areas, where files have been deleted, can no longer be used to store data. See “Restoring a Complete Backup,” below.
- A user accidentally deletes an important file or directory; see “Restoring Portions of an Archive Dataset,” below.

In other cases, you restore data because something is wrong with the disk or file system. The following situations can signal serious problems that require investigation; see “Recovering a Corrupted Volume,” later in this section.

- A disk suddenly becomes unusable.
- After a weekly backup, you are informed that the file system has become corrupted.
- I/O errors (error 301) occur on a disk. (These are recorded in the system log file, which you can read with the **PLog** command; see Section 19, “Troubleshooting”).

## Restoring a Complete Backup

Before you restore a complete backup to the disk from which it was taken (as you would to correct fragmentation), you must reinitialize the disk. See Section 11, "Adding Hard Disks," for information about initializing disks.

### Caution

Reinitializing destroys all data on the disk. Be sure that you have a current backup before you reinitialize a valid volume.

To restore a complete backup, follow these steps:

1. Insert the tape you want to restore into the tape drive.
2. On the Executive command line, type **Restore Archive**; then press **RETURN**.
3. Fill in the command form as shown in the following example. Parameter fields are described in Table 13-2. (See also the **Restore Archive** command in the *CTOS Executive Reference Manual* for more detailed parameter descriptions.)

Restore Archive

[Archive dataset ([QIC])]	[QIC]
[File list from (<*>*)]	<*>*
[File list to (<*>*)]	[d1]<*>*
[Overwrite ok?]	_____
[Confirm each?]	_____
[Sequence number]	_____
[Merge with existing file?]	_____
[List files only?]	_____
[Print file]	_____
[Suppress user interaction?]	_____
[Apply attributes?]	_____

4. Press **GO**.



**Table 13-2. Restore Archive Parameters**

Field Name	Description
<i>[Archive dataset ([QIC])]</i>	Default: <i>[QIC]</i>  Enter the tape device name containing the backup you want to restore.
<i>[File list from (&lt;*&gt;*)]</i>	Default: All files  If you leave this field blank, all files in the archive dataset are restored. To restore selected files, enter the directory specification and file name. (See also "Restoring Portions of an Archive Dataset," later in this section.)
<i>[File list to (&lt;*&gt;*)]</i>	Default: See below  If you leave this field blank, files are restored to their original directories on the default volume. To restore files to a volume other than the default, enter the volume or device name before the directory and file specification, for example, <i>[d1]&lt;*&gt;*</i> .
<i>[Overwrite ok?]</i>	Default: Prompt user  If you leave this field blank, you are informed when a file from the archive dataset already exists on the disk. To overwrite the disk file, enter <b>Yes</b> . To prevent overwriting, enter <b>No</b> .
<i>[Confirm each?]</i>	Default: No  If you leave this field blank, all specified files are restored without confirmation. To be prompted to confirm restoration of each file, enter <b>Yes</b> .
<i>[Sequence number]</i>	Default: 1  If you leave this field blank, restoration begins with the first tape of a multiple-tape set. To specify a tape other than the first, enter its sequential number in the set (for example, 2 or 3).

---

continued



**Table 13-2. Restore Archive Parameters (cont.)**

Field Name	Description
<i>[Merge with existing file?]</i>	<p>Default: No</p> <p>If you leave this field blank, and the tape contains unreadable data, corresponding disk data is overwritten and destroyed. To prevent corrupted tape data from overwriting existing disk data, enter <b>Yes</b>. (See also "Recovering a Corrupted Volume," later in this section.)</p>
<i>[List files only?]</i>	<p>Default: No</p> <p>To list the contents of an archive dataset without restoring it, enter <b>Yes</b>. If you leave this field blank, restoration takes place in the usual manner.</p>
<i>[Print file]</i>	<p>Default: Screen only</p> <p>To write command output to a file or printer (in addition to the screen), enter a printer or file specification. If you leave this field blank, command output is written to the screen only.</p>
<i>[Suppress user interaction?]</i>	<p>Default: No</p> <p>If you enter <b>Yes</b>, <b>Restore Archive</b> exits with an error when user interaction is required. This parameter is most frequently used in software installation scripts. If you leave this field blank, <b>Restore Archive</b> pauses when user interaction is required.</p>
<i>[Apply attributes?]</i>	<p>Default: No</p> <p>If you enter <b>Yes</b>, Access Control attributes are applied to the restored files as they existed on the volume from which the backup was taken. If you leave this field blank, Access Control attributes are not applied. (Access Control is a separately installable system security product.)</p>

### Restoring Portions of an Archive Dataset

If necessary, you can restore a single file, an entire directory, or a group of files. You might need to do this, for example, if a user accidentally deletes an important file.

The following procedure describes how to restore a single file. Use wildcard characters or an at-file to restore groups of files or a directory (see “\* and ? Wild-Card Characters,” in the *CTOS Executive User’s Guide* or “Wild-Card Characters” in the *CTOS Executive Reference Manual*).

1. Insert the archive tape into the tape drive.
2. Start the **Restore Archive** command with the Executive. (See the preceding procedure, “Restoring a Complete Backup.”)
3. Fill in the form, as shown in the following example. (See Table 13-2 for parameter field descriptions.)

Restore Archive

[Archive dataset ([QIC])]	<u>[QIC]</u>
[File list from (<*>*)]	<u>&lt;Dir&gt;FileNme</u>
[File list to (<*>*)]	<u>[d0]&lt;Dir&gt;FileNme</u>
[Overwrite ok?]	<u>_____</u>
[Confirm each?]	<u>_____</u>
[Sequence number]	<u>_____</u>
[Merge with existing file?]	<u>_____</u>
[List files only?]	<u>_____</u>
[Print file]	<u>_____</u>
[Suppress user interaction?]	<u>_____</u>
[Apply attributes?]	<u>_____</u>

4. Press **GO**. You are informed when the file has been restored.

### Recovering a Corrupted Volume

At some point during your career as a system administrator, you will most likely encounter a *corrupted volume*. This means that data on the disk contains unreadable or unintelligible errors. Such errors can be caused by both hardware and software problems.

This section contains the following information to help you identify and correct corrupted volumes:

- How to identify a corrupted volume
- How to back up a corrupted volume
- How to troubleshoot disk problems
- How to restore data

### Identifying a Corrupted Volume

Signs and symptoms of a corrupted volume vary, depending on the type and extent of damage. In some cases, the disk itself is physically damaged; in unusual cases, software errors can garble the file system.

To check for physical damage to the disk, use the **PLog** command to read the system error log. It records new bad spots that result from accidents or normal wear and tear on the disk. See Section 19, "Troubleshooting," for detailed information about the **PLog** command.

To check for software errors, a file system verification takes place when you perform volume backups. If problems are detected, you are informed that the volume has been corrupted. You can also run a file system verification independently of a backup, at any time you suspect disk problems; see the **Volume Archive** command in the *CTOS Executive Reference Manual*.

In addition, error messages can occur when you execute commands. The following error message signals potential problems:

```
I/O error (Error 301)
```

Do not ignore I/O error messages. Back up the disk immediately; then repair or replace the disk.

### Backing Up a Corrupted Volume

Fortunately, most corrupted volumes can be almost completely recovered with the backup and restoration procedures in this section. If, however, the volume home block (VHB) is corrupted, the operating system will not recognize the disk as a valid CTOS volume. If this should happen, you will be prompted to supply the following information during the backup procedure:

Device password	Enter the device password assigned to the disk. For prebuilt operating systems, the device password matches the device name.
Disk type	For workstation modules, enter the disk-type vendor code. For SRP disks, enter the device template listed in Table 11-4.

See Section 11, "Adding Hard Disks," for more information about device passwords and disk types. See the **Volume Archive** command in the *CTOS Executive Reference Manual* for more information about backing up corrupted volumes.

### Troubleshooting Disk Problems

After you have backed up a corrupted disk, it is important to correct the problem before you restore data to it. If possible, install a spare disk and restore the backup to it while the original disk is being repaired. If this is not possible, try to prevent users from continuing to use a questionable disk by physically removing it, or warn them that they may lose data. Then, perform the following troubleshooting steps before putting the disk back into regular use.

**Caution**

The following procedure destroys all data on the disk.

1. Run hardware diagnostics on the disk, or follow your usual procedure to obtain service.

2. If the disk passes diagnostics, reinitialize the disk (see Section 11, "Adding Hard Disks"); be sure to run at least eight surface tests on the disk. If possible, initialize the disk overnight and run a large number of surface tests.

## Restoring Data

After you have repaired or replaced a corrupted volume, you can usually recover most of the data by restoring your routine backups, as well as the backup you obtained after problems occurred. To recover the maximum amount of data, restore the archive tapes as follows:

1. Restore the most current complete volume backup you performed before disk problems occurred. (Use the procedure for restoring a complete backup, earlier in this section.)
2. Restore the incremental backups you performed after the most recent complete volume backup. (Use the procedure for restoring a backup, earlier in this section.)
3. Restore the backup you obtained from the corrupted volume. Follow the procedure for restoring a complete backup; however, specify **Yes** in the *[Merge with existing file?]* field, as shown below. This protects files that have been corrupted since you performed the routine backups.

### Restore Archive

[Archive dataset ([QIC])]	<u>[QIC]</u>
[File list from (<*>*)]	<u>&lt;*&gt;*</u>
[File list to (<*>*)]	<u>[d0]&lt;*&gt;*</u>
[Overwrite ok?]	<u>yes</u>
[Confirm each?]	<u></u>
[Sequence number]	<u></u>
[Merge with existing file?]	<u>yes</u>
[List files only?]	<u></u>
[Print file]	<u></u>
[Suppress user interaction?]	<u></u>
[Apply attributes?]	<u></u>





## Section 14

# Using a File-System Cache

## What Is a File-System Cache?

A file-system cache is an area of memory where files are stored dynamically, as they are used. When a file is cached, it is accessed from memory, rather than from disk, which increases file access speed.

The file-system cache is not a RAM disk, although it can be used as one. In most cases, you do not specify particular files to be stored in the cache. Files circulate in and out of the cache as they are used. Therefore, the file-system cache increases performance for many frequently used files rather than a preselected few.

## How Caching Works

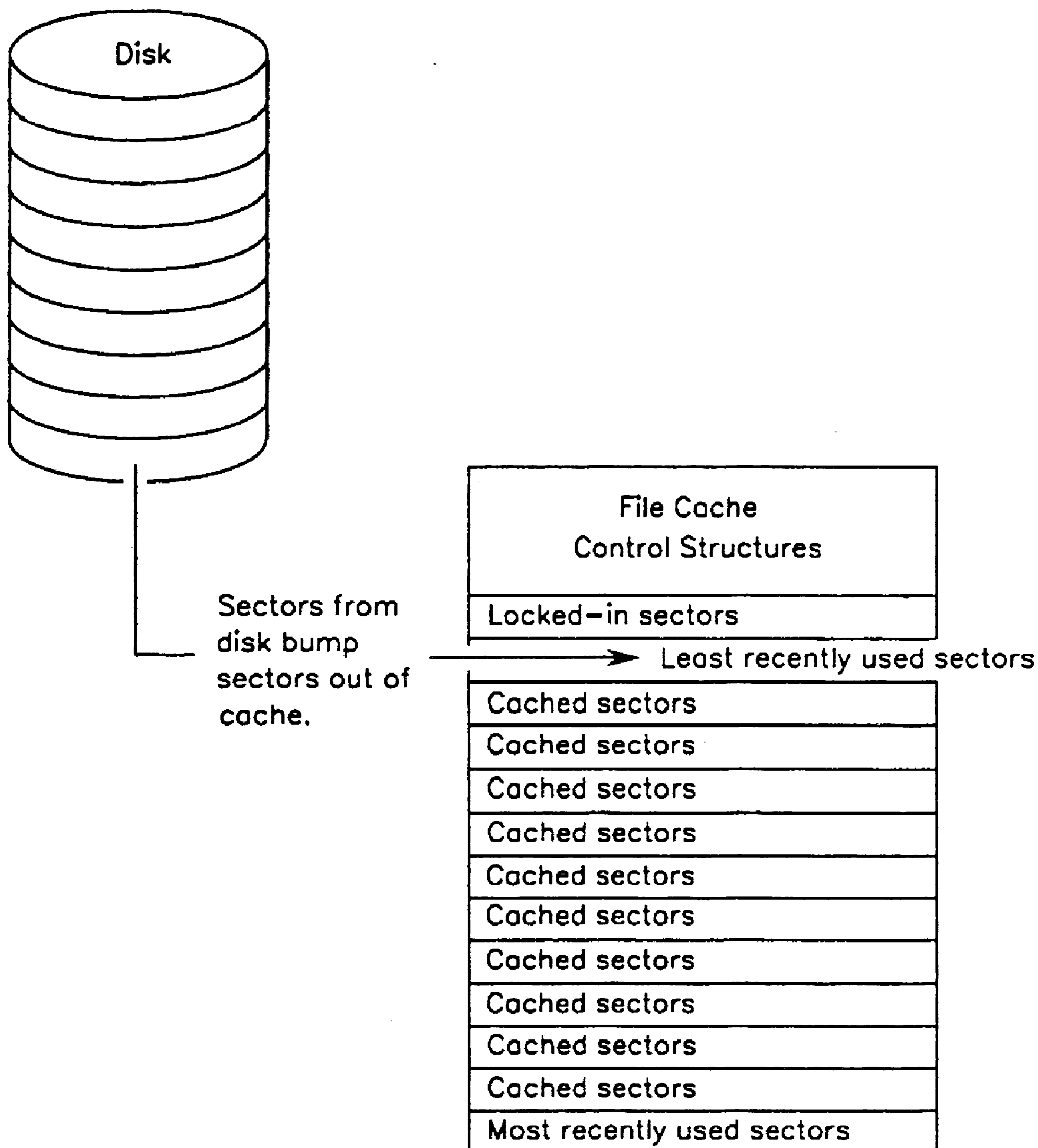
When you access a file, sectors from it are read into the file-system cache. Then, as you continue to use those sectors, they are accessed from the cache, rather than from disk. This provides faster access to the file.

Because the file-system cache is limited in size, however, it can accommodate only a certain number of disk sectors. Therefore, sectors that have not been recently used are “bumped” from the cache, while the most recently used sectors remain in the cache. You can also tag certain files to be locked into or out of the cache. Those options are described later in this section.

A file-system cache is depicted in Figure 14-1. Notice that a portion of the cache contains cache control structures. Therefore, not all memory that you allot to the cache is available for storing files.

If the cache becomes full, you may notice a decrease in file access speed. You can check cache usage with the **Cache Status** command; see the *CTOS Executive Reference Manual* for information about using that command. By monitoring the file-system cache, you can determine whether a larger cache could be of benefit to your system.

Figure 14-1. File-System Cache



603.14-1

## Write-Through Caching

The default method of file-system caching is called *write-through* caching, where changes to cached sectors are immediately written to disk, as well as to the cache. Write-through caching is equally efficient for both read and write operations.

## Write-Behind Caching

A file caching option on virtual-memory systems is called *write-behind* caching, where changes to cached sectors are allowed to accumulate before they are written to disk. This speeds up file system access but poses a risk of losing data in cases of system crash or power failure. You can, however, configure a write-behind cache to predesignate the amount of data that is allowed to accumulate before it is saved back to disk.

For information about configuring a write-behind cache, see the *:FileCacheDefault:* and *:FileCacheService:* parameters, in Section 16, “Configuring Operating Systems.”

## Configuring Cache Memory

On a single-processor workstation (CTOS II or CTOS III only) and on the master processor of a multiprocessor system, a default cache of 500K bytes is allotted.

Memory to be allotted for the cache is defined by the following entries in the operating system configuration file (*Config.sys* on workstations or *SrpConfig.sys* on SRPs):

```
:FileCacheService:      (BlockSize = 4096
                          BlockCount = 128
                          MinWorkingSetBlockCount = 16
                          DirtyBlockCount = 10)
```

The default, as shown above, allots a 500K-byte cache, as follows.

4096	(size of each block in bytes)
x 128	(number of blocks)
524,288	(total number of bytes allotted)

## Using a File-System Cache

---

To change the size of the cache, increase or decrease the value of *BlockCount*. The following example allots a 2M-byte cache.

```
:FileCacheService:      (BlockSize = 4096
                        BlockCount = 512
                        MinWorkingSetBlockCount = 16)
                        DirtyBlockCount = 10)
```

In most cases, you do not need to change the values of *BlockSize* or *MinWorkingSetBlockCount*, and *DirtyBlockCount* is for write-behind caching only. For more information, see the *:FileCacheService:* parameter, in Section 16, “Configuring Operating Systems.”

To allot cache memory on an IOP or an SRP processor other than the master processor, simply include the *:FileCacheService:* parameter for that processor in the *Config.sys* or *SrpConfig.sys* file.

## Setting File Attributes for Caching

When a workstation or SRP boots, a default caching attribute is assigned to all files with the *:FileCacheDefaultEnable:* parameter in the operating system configuration file. If the default is set to **Yes**, all files are enabled for future caching until caching is specifically disabled on a per-file basis. Likewise, if the default is set to **No**, no file is enabled for caching until caching is specifically enabled, again on a per-file basis. The procedure for setting the caching attribute on individual files is described below.

## Disabling Files for Caching

If the *:FileCacheDefaultEnable:* parameter is set to **Yes**, you can disable caching on individual files with the **Disable Caching** command, as shown in the following example:

```
Disable Caching
File list      FileName
[Print file]   _____
```

To reenabling caching, use the **Enable Caching** command; see the *CTOS Executive Reference Manual*.

You can also disable caching on selected files during system initialization. To do so, use the JCL Command statement, and specify the names of the files to be excluded. Enclose file names in parentheses and separate them with commas, as shown in the following example:

**Command Disable Caching, (*FileName1, FileName2*)**

Or, you can specify an at-file containing the list of files to be excluded. To do so, surround the at-file name in single quotation marks ('), as shown in the following example:

**Command Disable Caching, '@*FileName*'**

The **Disable Caching** command does not remove a currently cached file from the file-system cache. To do that, execute the **Unlock Cache** command on the file after you disable caching. See the *CTOS Executive Reference Manual* for information about using that command.

### Enabling Files for Caching

If the *:FileCacheDefaultEnable:* parameter is set to **No**, you can enable caching on selected files with the **Enable Caching** command. You can also enable caching on selected files during system initialization.

Procedures for enabling caching are identical to those described for disabling caching. See "Disabling Files for Caching," earlier in this section.

### Using the Cache as a RAM Disk

To emulate a RAM disk function, you can lock files into the cache. Locked-in files remain in the cache until they are specifically unlocked and removed. Remember, however, that the purpose of the cache is to improve general performance of all file-access operations; by locking many files into the cache, you could defeat that purpose.



## Using a File-System Cache

---

To lock files into the cache during system initialization, use the JCL Command statement, as shown in the following example:

**Command Lock In Cache, (*FileName1, FileName2*)**

Or, specify an at-file containing the list of files to lock in, as shown in the following example:

**Command Lock In Cache, '@*FileName*'**

Alternatively, you can lock files into the cache with the **Lock In Cache** command, as shown in the following example:

```
Lock In Cache
File list      FileName
[Print file]
```

To unlock a file and remove it from the cache, use the **Unlock Cache** command; see the *CTOS Executive Reference Manual*.

## Caching From Client Workstation to Server

The file-system cache, as described above, is configured for local caching only. For a client workstation to cache files located on the server, the *agent cache* must be enabled on the client workstation. To enable the agent cache, specify **Yes** to the following *Config.sys* parameter:

**:AgentCacheDefaultEnable: Yes**

When the agent cache is enabled, files that you open on the server are cached in memory on the client workstation. This greatly improves access speed to those files.

The caching utilities described earlier in this section work the same with the agent cache as they do with local file-system caching. Those cache utilities consist of the **Cache Status, Disable Caching, Enable Caching, Lock In Cache,** and **Unlock Cache** commands.



## Caching on Cluster IOPs

When client workstations communicate with a multiprocessor workstation server, they use the file-system cache on the server's master processor unless the agent cache is enabled on the Cluster IOP. To enable the agent cache, configure a cache on the IOP and specify **Yes** to the `:AgentCacheDefaultEnable:` parameter, as shown in the following example:

```
:Processor: EP01
:FileCacheService: (BlockCount = 258)
:AgentCacheDefaultEnable: Yes
```

Cluster access speed is not affected when agent caching is enabled; however, it frees the cache on the master processor for local processing activity.

## Caching on Diskless Workstations

The agent cache parameter must be enabled for generalized caching to take place from the server to a diskless workstation. It is possible, however, to lock files into the cache on a diskless workstation without enabling generalized caching. For example, you could lock in a frequently used application run file, such as the Executive or OFIS Document Designer. See "Using the Cache as a RAM Disk," earlier in this section.

## Remote Caching on an SRP

The term *remote caching* applies to SRPs only and refers to a disk using a cache on a processor different from the one to which it is physically connected.

## Using a File-System Cache

---

Remote caching parameters are contained in the operating system configuration file, *[Sys]<Sys>SrpConfig.sys*. To configure a remote cache, add the following entries after the *:FileCacheService:* entry, in the order shown below:

*:RemoteCachePool:* Designates the local cache, or sets up a new cache, for use by remote client processors.

*:RemoteCacheClient:* Identifies a particular remote processor as a caching client.

Examples of remote caching parameters are included below. See also the remote caching parameters, in Section 16, "Configuring Operating Systems."

## Sharing a Cache

For real-mode processors to share a cache on a protected-mode processor, they must be named as remote-cache clients. The following example shows FP00 and FP01 sharing the cache on GP00:

```
:Processor: GP00
  :FileCacheService: (BlockSize = 4096,
                    BlockCount = 512)
  :RemoteCachePool: (Name = Local)
  :RemoteCacheClient: (Name = FP00,
                     Pool = Local)
  :RemoteCacheClient: (Name = FP01,
                     Pool = Local)
```

First, the *:FileCacheService:* parameter allots a 2M-byte cache (4096 bytes times 512 equals 2,097,152 bytes). Then, *:RemoteCachePool:* designates the local cache on GP00 to be accessible to remote processors. (Note that the local cache on a protected-mode processor is always named "Local".) Finally, *:RemoteCacheClient:* names FP00 and FP01 as remote clients of the cache.

## Configuring a Remote Cache

In most cases, caching is most efficient when a real-mode processor shares a local cache on a protected-mode processor. To adjust for the increased workload imposed by cache sharing, you can make the cache size larger than the default, as shown in the preceding example. It is possible, however, to set up a remote cache for exclusive use by a specific remote processor. To do this, you allot a cache in addition to the local cache on a protected mode processor. The following example shows a local cache on GP00 and a remote cache for use by FP00.

```
:Processor:  GP00
  :FileCacheService:  (BlockSize = 4096,
                      BlockCount = 128)
  :RemoteCachePool:  (Name = c0
                     BlockSize = 4096,
                     BlockCount = 128)
  :RemoteCacheClient: (Name = FP00,
                      Pool = c0)
```

Note that the *:FileCacheService:* parameters allot a 500K-byte local cache on GP00. (Because that local cache will not be accessed by remote processors, however, it does not need to be identified with the *:RemoteCachePool:* parameter.) The *:RemoteCachePool:* parameter then sets up another 500K-byte cache named *c0*. Finally, the *:RemoteCacheClient:* parameter names FP00 as the remote client for the cache named *c0*.

Remote cache names can be anything other than a reserved device name or "Local", which is reserved for local caches. See the *CTOS Executive Reference Manual* for a list of reserved device names. The cache name *c0*, as shown in the preceding example, is a suggestion based on CTOS device-naming conventions.



# Section 15

## Optimizing System Performance

### Optimizing Partitioned Memory

Partitioned memory management is used by real-mode operating systems and protected-mode operating systems without virtual memory.

Partitions are contiguous areas of memory that are kept separate from one another. Processing that takes place in one partition does not interfere with processing in other partitions.

### Using Context Manager

Context Manager is an application that allows you to start and execute multiple applications concurrently on a single workstation. When you use Context Manager with partitioned memory, each application is started in its own partition.

The way in which you configure Context Manager can significantly affect system performance on partitioned memory systems, as described in this section. For detailed information about using Context Manager and its associated utilities, see the *CTOS Context / Window Manager Installation and Configuration Guide*.

### Calculating Partition Sizes

To determine minimum and maximum partition sizes for an application, you can use the **Version** command. In the *File list* field, specify the name of the application run file and in the *[Details?]* field, specify **Yes**, as shown in the following example:

```
Version
File list      [Sys]<Sys>RunFileName.run
[Details?]    Yes
[Print file]
[Run file mode]
[Min array, data, code]
[Max array, data, code]
[Keyboard ID]
[Alternate NLS Style?]
```

The information displayed by the **Version** command includes the following fields:

*Static memory size*

*Minimum dynamic memory size*

*Maximum dynamic memory size*

To determine the smallest partition size, in bytes, in which the application can run, add the values for static memory and minimum dynamic memory. You specify a relatively small partition size when it is important to keep a group of applications running without swapping in a small amount of memory.

To determine the largest partition size that will benefit performance, add the values for static memory and maximum dynamic memory. A partition that is larger than the static and maximum dynamic memory added together reserves memory that will never be allocated by the application and, therefore, consumes memory that could be available to other partitions.



### Using Static Partitions

*Static partitions* are created to the size specified in the Context Manager configuration file. On real-mode workstations, Context Manager creates static partitions only. On protected-mode workstations without virtual memory, the partition type (either static or variable) is specified in the Context Manager configuration file.

The use of static partitions ensures that a particular application runs in the size of partition you specify. This is useful for applications that require a large amount of memory to perform optimally. For example, using fonts or integrating objects with OFIS Document Designer is more efficient in a large partition.

To specify static partitions in the Context Manager configuration file, type the partition size number, as shown in the following example:

*Memory required*    **500** KBytes

### Using Variable Partitions

On protected-mode workstations, you can specify either a static or a variable partition for each application. In the Context Manager configuration file, you specify a maximum size for a variable partition. Then, when Context Manager starts the application, it creates the variable partition to the maximum size specified, swapping other applications, if necessary, to obtain the contiguous memory. As other applications are swapped back into memory, however, the size of the variable partition may be reduced so that a maximum number of applications can run simultaneously, without swapping.

To specify a variable partition in the Context Manager configuration file, add a less-than sign (<) in front the partition-size number, as shown in the following example:

*Memory required*    **<500** KBytes

Optimal sizes for variable partitions are determined by your requirements. For example, you can minimize swapping by keeping maximum partition sizes small enough so that applications can be started with a minimum of swapping. Or, you can maximize performance of individual applications by specifying larger variable partition sizes.

### Code Sharing

Certain applications, such as the Executive, are designed to share code among partitions, if more than one instance of the application is running concurrently. To take advantage of code sharing, such applications must be started in variable, rather than static, partitions.

When partitions for code-sharing applications are viewed with the **Partition Status** command, the full partition size, including the shared code, is displayed for each instance of the application. For that reason, the total amount of memory displayed in the *Allocated* fields of the main display may exceed the actual amount of physical memory available on the processor. To determine how much memory is actually being consumed by code-sharing partitions, the amount of memory displayed in the *Code* field on the Details display should be subtracted from all but one of the code-sharing partitions.

Shared code remains available to each code-sharing partition, regardless of the order in which they were started, until the last code-sharing partition is exited.

### Optimizing Virtual Memory

Partition sizes specified in the Context Manager configuration file are not relevant to virtual memory operating systems. Virtual memory operating systems use *demand-paged memory management*, which loads applications into memory in 4K-byte *pages* and allocates memory in 4K-bytes *frames*, as it is needed.

When the total number of pages allocated to applications and system services is greater than the number of frames available, physical memory is said to be *oversubscribed*. In addition, to bring pages into physical memory, the paging service is said to process *page faults*.

Oversubscription of memory and page faults are normal and acceptable conditions. They make it possible for more applications than can be loaded into physical memory to run concurrently, usually with good performance. Problems can arise, however, if there is not enough physical memory for all the programs to run effectively.

### Suppressing Global Policy

In most cases, on virtual memory systems, swapping with Context Manager takes place as usual or is faster than on a partitioned memory system. In rare cases, however, a particular application consumes all available memory on the workstation. In such a case, an unusually long delay occurs when switching from that application back to Context Manager, and again when starting a new application.

If you experience long delays when switching or starting applications, you may be able to improve performance by suppressing the global paging policy. To do so, specify **Yes** to the `:SuppressGlobalPolicy:` parameter in `[Sys]<Sys>Config.sys`, the operating system configuration file. When the global paging policy is suppressed, memory for applications started with Context Manager is allocated according to the partition sizes specified in the Context Manager configuration file.

### Correcting Global Thrashing

A condition called *global thrashing* can degrade performance on virtual memory systems. This occurs when the number of frames required by concurrently running programs exceeds the amount of physical memory. To check for global thrashing, add together the amount of memory displayed for each program in the *Working set (cur)* field of the **Partition Status Details** display. If that amount exceeds the total amount of memory, as displayed in the *Total RAM* field of the **Partition Status** main display, you can improve performance by reducing the number of programs that are running concurrently.

For more information about the **Partition Status** command, see the *CTOS Executive Reference Manual*. For more information about memory management and the paging service, see the *CTOS Operating System Concepts Manual*.

# Allocating Queues

A *queue* is a portion of memory for storing a list of files or jobs awaiting processing. For example, when you initiate a spooled print request, it is stored in a queue until the file is printed. The Queue Manager maintains queues in an orderly manner so that jobs can be processed in a particular order. The order is either on a first-come, first-served basis, or according to user-specified priorities.

There are two types of queues:

- Dynamic queues, which are created when they are needed by an application
- Static queues, which are created when the Queue Manager is installed

The type of queue is determined entirely by the application that is using it. This information is available in the release documentation and manuals for the application. When configuring an application that requires queues, you will need to know what type of queues to install.

The following sections describe both dynamic and static queues. See the *CTOS Generic Print System Administration Guide* and the **Install Queue Manager** command in the *CTOS Executive Reference Manual* for information.

## Dynamic Queues

Dynamic queues are created when they are needed by an application. For example, when using the Generic Print System (GPS), queues are not created until a printer is installed. This means that you can add and remove queues without deinstalling the Queue Manager.

In addition, dynamic queues can take advantage of the Queue Manager's cache, which reads data from memory, rather than from a disk. By allotting more dynamic queues than you need, you increase the size of the cache. Although this requires more memory, it can enhance performance of the Queue Manager.



You allot dynamic queues when you install the Queue Manager. Use the following list to determine the number of dynamic queues to allocate:

- Allot one queue for each spooled printer.
- Allot one queue for each application that uses the Queue Manager, such as background Batch.
- Allot two additional queues to allow for expansion.

Optionally, if the system has enough memory, you can increase the size of the cache by allotting up to double the number of queues you need.

### Static Queues

Static queues are created when the Queue Manager is installed, regardless of whether they are eventually used. Information for creating static queues is contained in the *[Sys]<Sys>Queue.index* file. To add a static queue, you must deinstall the Queue Manager, edit *Queue.index*, and then reinstall the Queue Manager.

When you install the Queue Manager, space is automatically allocated for the static queues defined in *Queue.index*. See the *CTOS Generic Print System Administration Guide* for more detailed information about the *Queue.index* file.

### Optimizing Use of Disk Space

On many systems, the majority of disk input and output activity is performed on the system volume, while other disks remain idle. On a workstation, you can easily observe this situation. If the drive light is always on, the disk is being overworked.

In many cases, you can increase efficiency by distributing frequently used files among disks, as described in the following sections.

### Moving the “Scratch” Volume

Most applications create temporary files as you use them. Those files are stored on a scratch volume, which is designated by the operating system. By default, both workstation and SRP operating systems define *[Sys]* as the scratch volume. On protected-mode workstations and SRPs, however, you can reduce activity on *[Sys]* by redefining the scratch volume.

To redefine the scratch volume, edit the following field in the operating system configuration file:

*:ScratchVolumeName:*

When you change the scratch volume, you must also create a dollar-sign directory on the new scratch volume. Name this directory *<\$000>* and make it large enough to hold 750 files.

If a client workstation is booted locally, its scratch volume must be on one of its own local disks, rather than a disk on the server.

Client workstations that boot from the server may use either a local disk or the server’s system volume as the scratch volume. Diskless client workstations use the server’s scratch volume.

On SRPs, you can specify a scratch volume for each processor. Use this feature to assign one scratch volume per cabinet, to avoid interprocessor data transfers.

See Section 16, “Configuring Operating Systems,” for more detailed information about operating system configuration files.

### Moving Applications

Run files for applications are installed, by default, on the system volume. To reduce the workload on *[Sys]*, you can move some of the application run files to different disks. For example, Document Designer places a particularly heavy load on the system volume. By moving its run file to a different disk, you can speed operations that require disk activity.

When you move application run files, be sure to update user files, Context Manager configuration files, and command files that contain references to the application. Also be aware that some applications do not work correctly if they are not located in *[Sys]<Sys>*. See the application release documentation for that type of information.



## Conserving Memory on Single-Processor CTOS III Servers

The CTOS III server operating system provides for interprocessor communications on multiprocessor workstations. Since that capability is not needed on a single-processor server, you can conserve memory by installing a standalone operating system and then specifying the Server Agent in the operating system configuration file, as shown in the following example:

```
:SrvrAgentFile:[Sys]<Sys>SrvrAgent.run
```

That has the effect of installing a smaller operating system than the one required for multiprocessor servers, yet still providing the CTOS cluster communications capabilities.

See Section 3, "Understanding System Software," for information about the prepackaged CTOS III operating systems. See Section 16, "Configuring Operating Systems," for more detailed information about the *:SrvrAgentFile:* parameter.

## Optimizing Memory Usage on an SRP

In many cases, you can increase the efficiency of an SRP by redistributing processing among several processors. You can usually do this by rearranging system services and peripheral hardware or by reallocating memory blocks. The following sections describe some common methods of maximizing processing power.

### Isolating Disk-Intensive Applications

Some applications, such as ISAM or electronic mail, are disk intensive. This means that they frequently read and write disk data. To improve the performance of disk-intensive applications, isolate the appropriate system services on a dedicated disk-controlling processor (not the master processor). Then, copy all related directories and files to a disk controlled by the dedicated processor.

A disk-controlling processor is a General Processor with SCSI Interface, a File Processor, or a Data Processor.

### Moving Communications Services

If communications gateways are installed on a Cluster Processor, try moving them to a General Processor. If your SRP is not equipped with a General Processor, install communications gateways on a Terminal Processor or the Cluster Processor supporting the fewest client workstations.

### Avoiding Interprocessor Data Transfers

Whenever data is transferred between SRP processors, processing speed is reduced. This is particularly true for system services and commands executed via Cluster View. The following example demonstrates how interprocessor data transfers slow down processing.

Suppose your QIC tape drive is controlled by GP00 and you want to back up a disk also controlled by that processor. If you start a backup via Cluster View on GP00, data is processed and written to the tape by GP00. If, however, you start Cluster View on a different processor, for example, CP00, two interprocessor data transfers take place. First, data is read from the disk by GP00 and transferred to CP00 for processing; then it is transferred from CP00 back to GP00 to be written to tape.

System services that perform many disk operations perform best when installed on processors that control disks. For example, the Mail Service can be installed on any processor; installing it on a General Processor with SCSI Interface, a File Processor, or a Data Processor is the best choice, however, because disk operations are the most time consuming. In addition, the mail center should be configured to use a disk that is controlled by the processor on which the Mail Service is installed.

### Adjusting Memory Blocks

*Memory blocks* are used for interprocessor data transfers on workstations and SRPs. They affect performance as follows:

- Too few memory blocks reduce the speed of interprocessor data transfers.
- Too many memory blocks consume memory that could be made available for processing.

This section describes different types of memory blocks and provides guidelines for adjusting them. You allot memory blocks in the operating system configuration file (*[Sys]<Sys>Config.sys* on workstations and *[Sys]<Sys>SrpConfig.sys* on SRPs).

### What Are Blocks?

Blocks are small portions of memory that are allocated by the operating system. There are several types of blocks:

- *X-blocks* are used during communications between the server and client workstations.
- *W-, Y-, and Z-blocks* are used during data transfers between processors on multiprocessor systems.

Adjusting the size and number of blocks can enhance system performance in certain situations.

### X-Blocks

The prebuilt operating systems allocate enough X-blocks for most cluster environments. In the following situations, however, you may need to allot additional X-blocks:

- When a processor is supporting many diskless workstations
- When a processor is running a customized operating system that supports more workstations than the prebuilt version

You can display information about X-block usage on the server. To do so, start the **Cluster Status** command. Then press **F5 (Blocks)**.

The *Number XBlock waits* field is incremented by one each time cluster processing must wait for a free X-block. If the system frequently waits for X-blocks, performance could most likely be improved by increasing the number of X-blocks.

To allot more X-blocks, modify the *:XBlocks:* and *:XBlocksSmall:* entries in the operating system configuration file. The operating system uses small X-blocks whenever possible. If data is too large for a small X-block, or if all small X-blocks are being used, the operating system uses a large X-block.

### W-, Y-, and Z-Blocks

W-, Y-, and Z-blocks are used for interprocessor data transfers on multiprocessor workstations and SRPs. As with X-blocks, the prebuilt operating systems allocate enough W-, Y-, and Z-blocks for many environments. If, however, a processor is running many system services, you may need to allot more blocks to optimize performance.

The operating system transfers data in the smallest available block. The size of a data transfer is determined by the application. Few applications require W-blocks; therefore, the default number of W-blocks is 0.

To determine how W-, Y-, and Z-blocks are being utilized, use the **Multiprocessor Statistics** command, as described in the *CTOS Executive Reference Manual*.

To allocate more W-, Y-, and Z-blocks, modify the *:WBlocks:*, *:YBlocks:*, and *:ZBlocks:* parameters in the SRP operating system configuration file.

### Using a Cache Memory Disk

A *cache memory disk* is a read/write RAM disk. It is used as a scratch volume (*[Scr]*), which can improve the performance of some applications. A cache memory disk is, however, volatile memory. That means that if the system crashes or is rebooted, all files stored in the cache memory disk are lost. Because files stored on a scratch volume are dispensable temporary files, that is not a problem. As you can imagine, though, it would be a problem if the memory disk contained your working data files.

**Note:** *Cache memory disks are not supported on CTOS I/O processors.*

The following example shows how a cache memory disk is configured as a scratch volume in the operating system configuration file.

```
:MassStorage: (Class = CacheMemory,  
              Unit = 0,  
              Device = m0,  
              Volume = CMScratch,  
              MaxSectors = 4096,  
              MaxDirectories = 5,  
              MaxTempFiles = 700,  
              MaxSysFiles = 15,  
              MaxFiles = 50)
```



To define the memory disk as the scratch volume, you specify its volume or device name for the *:ScratchVolumeName:* parameter in the operating system configuration file, as shown in the following example:

```
:ScratchVolumeName: CMScratch
```

For more information, see the *:MassStorage:* parameter, in Section 16, “Configuring Operating Systems.”

When you use a cache memory disk as the scratch volume, you may need to create certain directories on it so that applications will work correctly. Such directories can include *<Spl>*, *<WP>*, and *<GPS>*; see the documentation for your applications for more detailed information.

The following example shows how you can set up your *SysInit.jcl* file to create directories during system initialization:

```
Command Create Dir,([Scr]<Spl>,[Scr]<GPS>,[Scr]<wp>)
```





# Section 16

## Configuring Operating Systems

### The Operating System Configuration File

The CTOS operating systems are designed to work well in a variety of settings. In some cases, however, you may need to configure an operating system to function optimally in your environment, for example, to set up a software development environment or to use hardware components that are not recognized by the operating system.

This section describes how to configure operating systems. The default operating system configuration files are as follows:

- *[Sys]<Sys>Config.sys* for workstations
- *[Sys]<Sys>SrpConfig.sys* for SRPs

On multiprocessor workstations and SRPs, the default configuration file contains information for a minimal hardware configuration, so in most cases, you will need to modify the configuration file for those systems.

### Configuration File Format

Each line of the operating system configuration file is written in the following format:

*:Keyword:Value*

where

*:Keyword:* Is the name of a parameter.

*Value* Is the configurable value of a parameter.

In addition, the following construction pertains to parameters that consist of a keyword and subparameters:

*:Keyword: (Subparam = Value, Subparam = Value)*

where

<i>:Keyword:</i>	Is the name of a parameter.
<i>Subparam</i>	Is the name of a subparameter.
<i>Value</i>	Is a configurable value for a parameter or subparameter.

### Configuration File Syntax

The following rules of syntax apply to parameters and subparameters:

- The colon preceding each keyword must be the first character on the line; no spaces or characters may precede it.
- A space after the colon following a keyword is optional.
- Parentheses enclose the entire set of subparameters.
- A space or an equal sign (=) separates a subparameter from its value.

If subparameters do not fit on a single line, they may continue on the next line, as shown below:

```
:Keyword:      (subparam = value,  
                  subparam = value,  
                  subparam = value)
```

The amount of white space between keywords, subparameters, and values is not significant. In this manual, white space is used to enhance the readability of the configuration file.

## Configuration Files for Single-Processor Workstations

A sample configuration file for a single-processor workstation is shown in Figure 16-1. It consists of a single, unlabeled section, which contains all parameters that pertain to the workstation.

**Figure 16-1. Operating System Configuration File for a Single-Processor Workstation**

---

```
:CrashDumpFile:[Sys]<Sys>CrashDump.sys
;ExtCrashDumpFile:[Sys]<Sys>ExtCrashDump.sys
:SwapFile:[Sys]<Sys>ExtCrashDump.sys
:ScreenTimeout:30
:LFSToMaster:[Sys]<Sys>*
:FileCacheService:(BlockSize = 4096, BlockCount = 128)
:FileCacheDefaultEnable:Yes
:AgentCacheDefaultEnable:Yes
```

---

# Configuration Files for Multiprocessor Systems

A sample configuration file for a multiprocessor workstation is shown in Figure 16-2 and for an SRP in Figure 16-3. The arrangement of configuration file entries for multiprocessor systems differs from that of a single-processor workstation in that it contains a boot section and a processor section for each processor. Those sections are described in detail below.

**Figure 16-2. Operating System Configuration File for a Multiprocessor Workstation**

---

```
:Boot: (Processor = EP01, OS = [Sys]<Sys>vSrvr.img)
:ScreenTimeout:10

:Processor: Default
:ClusterLineSpeed: 1.8M
:XBlocks: (Number = 32)
:YBlocksIn: (Number = 32)
:YBlocksOut: (Number = 32)

:Processor: EP00
:FileCacheService: (BlockCount = 512)
:FileCacheDefault: (Enable = Yes, WriteBehind = Yes)

:Processor: EP01
:ClusterLineSpeed: 3.7M
:XBlocks: (Number = 64)
:FileCacheService: (BlockCount = 258)
:AgentCacheDefaultEnable:Yes
```

---

**Figure 16-3. Operating System Configuration File for an SRP**

```

:LogUnknownEntries: Yes

:Boot: (Processor = GP00, Dump = No)
:Boot: (Processor = GP00, Dump = No)
:Boot: (Processor = CP00, OS = [Sys]<Sys>rSrpCp.run, Dump=No)
:Boot: (Processor = CP01, OS = [Sys]<Sys>rSrpCp.run, Dump = No)
:Boot: (Processor = FP00, OS = [Sys]<Sys>rSrpFp.run, Dump = No)
:Boot: (Processor = TP00, OS = [Sys]<Sys>rSrpTp.run, Dump = No)
:Boot: (Processor = SP00, OS = [Sys]<Sys>rSrpSp.run, Dump = No)
:Boot: (Processor = GP01, OS = [Sys]<Sys>pSrpGp.img, Dump = No)
:Boot: (Processor = DP00, OS = [Sys]<Sys>rSrpDp.run, Dump = No)

:Processor: Default
:ClusterLine1: (Speed = 1.8M)
:ClusterLine2: (Speed = 1.8M)
:CrashDumpPath: [Sys]<CrashDump>

:Processor: GP00
:WatchDogStatus: SetFlag
:ClusterLine2: (Speed = 3.7M)
:FileCacheService: (BlockSize = 4096, BlockCount = 512)
:MassStorage: (Class = SCSI, Adaptor = 0, Target = 1,
               LUN = 0, Device = d1, Password = d1)
:MassStorage: (Class = SCSI, Adaptor = 0, Target = 2,
               LUN = 0, Device = d2, Password = d2)
:MassStorage: (Class = SCSI, Adaptor = 0, Target = 3,
               LUN = 0, Device = d3, Password = d3)
:SequentialStorage: (Class = SCSI, Adaptor = 0,
                    Target = 0, LUN = 0, Device = QIC)
:XBlocks: (Number = 64)
:ZBlocks: (Number = 200)

:Processor: CP01
:ClusterLine1: (Speed = 307K, MaxWs = 4)

:Processor: FP00
:MassStorage: (Class = ST506, Unit = 1, Device = d4, Password = d4)
:MassStorage: (Class = ST506, Unit = 2, Device = d5, Password = d5)

:Processor: GP01
:FileCacheService: (BlockSize = 4096, BlockCount = 3000)
:RemoteCachePool: (Name = Local)
:RemoteCacheClient: (Name = FP00, Pool = Local)
:RemoteCacheClient: (Name = DP00, Pool = Local)

:Processor: DP00
:MassStorage: (Class = SMD, Unit = 0, Device = s0, Password = s0)
:MassStorage: (Class = SMD, Unit = 1, Device = s1, Password = s1)
:SequentialStorage: (Class = HalfInch, Unit = 1, Device = Tape)

```

---

### Boot Section

The boot section identifies the processors and operating systems to be booted by the master processor. It consists of the following keywords and subparameters:

*:Boot: (Processor=Xpnn, OS=FileSpecification, Dump=Yes/No)*

Subparameters are described below.

Subparameter	Value
Processor	Specify the four-character processor identifier, for example, <b>GP01</b> .
OS	Specify the file specification for the operating system to be booted on the processor, as shown in the following example:  <b>[Sys]&lt;Sys&gt;pSrpGp.img</b>
Dump	Specify either <b>Yes</b> or <b>No</b> . This subparameter defines whether a memory dump takes place automatically when the processor crashes.

The boot section precedes any processor entries in the configuration file. (On SRPs only, the *:LogUnknownEntries:* parameter may precede the boot section, as shown in Figure 16-3. See its description under “Configurable Parameters,” later in this section.)

**Note:** *The master processor boots from [Sys]<Sys>SysImage.sys; therefore, any file specification value in the :Boot: list for that processor is ignored. You can, however, include a Dump subparameter for the master processor, for example, :Boot: (Processor=GP00, Dump=Yes)*

### Processor Section

The processor section contains default parameter values, as well as specific parameters for individual processors. The processor-entry format is described below. Configurable parameters are described in detail later in this section.

Processor entries are written in the following format:

*:Processor: Xpnn*

where *Xpnn* is the four-character processor ID. (See also Section 2, “Understanding Hardware.”)



### Processor Defaults

The first entry in the processor section defines default parameters. Processor default entries are optional; however, they can reduce the number of parameters that you need to specify for individual processors.

To define processor defaults, specify **Default** in place of a processor ID, as shown below:

```
:Processor: Default
```

Beneath that entry, you define parameters, as shown in the following example:

```
:Processor: Default
  :YBlocks: (Number = 20, Size = 2560)
  :ZBlocks: (Number = 40, Size = 180)
  :ClusterLine1: (Speed = 1.8M, MaxWs = 8)
  :ClusterLine2: (Speed = 1.8M, MaxWs = 8)
  :XBlocks: (Number = 15, Size = 2624)
```

As each processor is booted, parameters that do not apply to it are ignored. For example, cluster line parameter values are implemented on Cluster Processors and General Processors only.

*Note:* If values unassociated with a *:Processor:* entry occur before the processor default section, those values are implemented on the master processor only.

### Entries for Specific Processors

After the default processor section, you define parameters for individual processors. Parameters for specific processors override default parameters.

In the following example, the cluster line parameters specified for GP00 override those that were set as processor defaults.

```
:Processor: GP00
  :ClusterLine1: (Speed = 3.7M, MaxWs = 16)
```

### Editing the Configuration File

To make changes to the operating system configuration file, follow these steps:

1. On the Executive command line, type **Editor**; then press **RETURN**.
2. Type the file name of the configuration file, as shown in the following example:

```
[File name(s)]  [Sys]<Sys>Config.sys
```

3. Press **GO**.

The configuration file appears on the screen. It should look similar to one of the samples shown earlier in this section.

4. Add keywords and edit values as required.
5. When you have finished making changes, press **FINISH**, then **GO**, to save the file.
6. Reboot the workstation or SRP.

See Section 4, "Using Administrative Tools," for more information about using the Editor.

### Creating *WsNNN>Config.sys*

You can create special versions of the operating system configuration file for client workstations that boot from the server. Such files must be located in *[Sys]<Sys>* on the server and are named as follows:

```
WsNNN>Config.sys
```

where *NNN* is a three-digit workstation-type number .

If *WsNNN>Config.sys* does not exist for a particular workstation-type number, the following file, if it exists, is used:

```
Ws>Config.sys
```

If neither *WsNNN>Config.sys* nor *Ws>Config.sys* exists, the workstation uses the server's version of *Config.sys*. If *Config.sys* does not exist (for example, on an SRP server), default parameters are used.

See also “Workstation-Type Numbers,” in Section 5, “Bootstrapping,” for more information.

*Note:* On CTOS III workstations only, the preceding information also applies to HwNNN>Config.sys. See Section 5, “Bootstrapping,” for more information.

## Using Special-Purpose Configuration Files

You can create special-purpose configuration files for use in different circumstances, for example, to establish an environment for software testing or troubleshooting.

On workstations, the default operating system configuration file, [Sys]<Sys>Config.sys, is used unless another file is specified with the **Bootstrap** command. On SRPS, a sequence of default files exist, depending on the keyswitch position from which the SRP is booted. Both situations are described below.

### Specifying a Configuration File With the Bootstrap Command

On CTOS III workstations, you can specify an operating system configuration file in the **Bootstrap** command form, as shown in the following example:

```

Bootstrap
  [File name]
  [Sys volume or WsNNN]
  [OS configuration file]  [Sys]<Sys>AltConfig.sys
  
```

### Associating Configuration Files With SRP Keyswitch Positions

On an SRP, the position of the keyswitch on the front of primary cabinet determines the configuration file that is read when the SRP boots.

The keyswitch-controlled files are named as follows:

[Sys]<Sys>SrpConfig.k.sys

where *k* corresponds to the keyswitch position.

In the absence of a configuration file for a particular keyswitch position, the SRP reads *[Sys]<Sys>SrpConfig.sys*.

See “Keyswitch Positions,” in Section 5, “Bootstrapping,” for a description of the SRP keyswitch.

## Configurable Parameters

Configurable parameters are described below in alphabetical order. You do not need to specify every parameter, and if you omit a parameter, a default value is used. In addition, all parameters do not necessarily pertain to every type of processor or operating system; relevant information is listed for each parameter.

### ***:ActionKeySticks:***

CTOS I, CTOS II, CTOS III

Default: No

This parameter specifies a default condition for the **ACTION** keystroke. Specify **Yes** if you want the keyboard to ignore the upstroke (release) of the **ACTION** key. This is used in cases where the user is physically incapable of holding down the **ACTION** key and another key simultaneously. The action key is deactivated when another key is pressed.

### ***:AgentCacheDefaultEnable:***

CTOS II and CTOS III; client workstations and Cluster IOPs only

Default: No

This parameter sets the default caching attribute for files opened from client workstations on disks located on the server. It also enables caching on multiprocessor-workstation Cluster IOPs. Specify **Yes** if you want to enable agent caching. See also Section 14, “Using a File-System Cache,” for a detailed description of the agent cache. If you specify **No** or leave this field blank, the agent cache is not enabled. Note that this parameter must be set to **Yes** for caching to take place on diskless workstations.



***:BeepOnToggle:***

CTOS I and CTOS II

Default: No

This parameter activates an audible tone on chord keys that toggle on and off (**SHIFT LOCK** on all keyboards and **NUM LOCK** on some) in applications that read the keyboard in character mode. When set to on, one long and one short beep sound when a key toggles on, and one long and two short beeps sound when it toggles off.

***:BAttrBlinking:***

***:BAttrBold:***

***:BAttrHalfBright:***

***:BAttrHalfReverse:***

***:BAttrNormal:***

***:BAttrReverse:***

***:BAttrStruck:***

***:BAttrUnderline:***

VAM bit-mapped display mode

Defaults: See below

These parameters define bit-mapped character attributes. You can specify **Normal**, **Halfbright**, **Underline**, **Reverse**, **Outline**, **Bold**, **Struck**, or **LowContrast**. Default values are as indicated by the keywords except for *:BAttrBlinking:*, the default of which is outlined text, and *:BAttrHalfReverse:*, which is described below.

The following examples demonstrate how to change attributes with the values listed above. To change the half-bright attribute to low contrast, make the following change in *Config.sys*:

*:BAttrHalfBright: LowContrast*

You can assign more than one attribute to a single video option. For example, to change the reverse screen attribute to half-bright reverse video, make the following entries in *Config.sys*. Notice that you must explicitly specify reverse video; it is implemented as the default value only if no other entry exists.

*:BAttrReverse: Halfbright*

*:BAttrReverse: Reverse*

To change half-bright reverse video to low-contrast, outlined reverse video, make the following entries:

*:BAttrHalfReverse:* LowContrast

*:BAttrHalfReverse:* Outline

*:BAttrHalfReverse:* Reverse

### ***:Boot:***

CTOS III (multiprocessor servers only) and CTOS XE

See “Boot Section,” earlier in this section.

### ***:CheckDAI:***

CTOS II and CTOS III

Default: No

Specify **Yes** to activate the device address identification (DAI) number. A DAI number is a physical ID that allows a program to identify a particular workstation within a workstation group.

Note that a DAI number requires special hardware and is not the same as a hardware ID. By default, hardware ID support is present in the operating system but is disabled if this parameter is set to **Yes**.

### ***:ChordKeysStick:***

CTOS I, CTOS II, and CTOS III

Default: No

This parameter causes the keyboard to ignore the upstroke (release) of chord keys (**SHIFT**, **CODE**, and **ALT**). It is used in cases where the user is physically incapable of holding down a chord key and another key simultaneously. The chord key is deactivated when the next nonchord key is pressed.

Note that this parameter also controls the **ACTION** key, but can be overridden by positioning the *:ActionKeySticks:* parameter below it in the configuration file.



***:ClientAgentFile:***

CTOS III

Default: None

This parameter specifies the name of the file that contains the loadable LAN software for CTOS III client workstations. It can be used to load a different LAN protocol from the standard CTOS cluster. Specify the file specification of the file containing the LAN software you want to use. See also *:SrvrAgentFile:*, later in this section.

**Note:** *The client agent for CTOS is in a file named Clster.sys and is copied to [Sys]<Sys> when you install the operating system software.*

***:ClusterLine1:*** (Speed = bps, MaxWs = number)

***:ClusterLine2:*** (Speed = bps, MaxWs = number)

CTOS II, CTOS III, and CTOS/XE; servers only; processors with cluster lines only

Default: See subparameters, below

**Note:** *:ClusterLine2: is for SRP processors only.*

This parameter defines the line speed and number of workstations supported on cluster communications channels. In most cases, maximum cluster line speed is dictated by the workstation hardware connected to the line. Check your workstation hardware specifications, and see the *CTOS Cluster and Network Hardware Installation Guide* for information about supported cluster line speeds.

Subparameters are described below.

Subparameter	Value
Speed	Default: 307kbps for SRPs; 1.8Mbps for workstations  Specify <b>307k</b> , <b>1.8M</b> , or <b>3.7M</b> ("k" means kilobits per second; "M" means megabits per second).
MaxWs	Default: 16 for SRP General Processors, 8 for SRP Cluster Processors, 32 for workstations, and 16 for Cluster IOP cards.  Specify the maximum number of workstations to be connected to the cluster line. To disable a cluster line, specify <b>0</b> .

### ***:ClusterLineSpeed:***

CTOS I; servers only

Default: 307kbps

**Note:** For CTOS II and CTOS III, this parameter is replaced by `:ClusterLine1:`, as described above.

Specify **3.7Mbps**, **1.8Mbps**, or **307kbps**, depending on the hardware configuration of the cluster. See the *CTOS Cluster and Network Hardware Installation Guide* for information about supported cluster line speeds. Client workstations use the line speed defined by the server.

### ***:ClusterTimeout:***

CTOS I, CTOS II, and CTOS III

Default: 30

Minimum: 4

Maximum: 65535

This parameter controls the amount of time that elapses before a client workstation terminates attempts to communicate with the server. Specify a number of seconds. If communication does not take place during the specified interval, the client workstation returns an error, as shown in the following example:

```
Cluster not running (Error 6)
```

The message can vary, depending on the cause of the communication error.

### ***:ColorMonitor:***

VAM; VGA only

Default: Yes

This parameter controls color or gray scale configurations on VGA hardware that is not recognized by the operating system. Specify **Yes** if color is not displayed on a color monitor. Specify **No** if an unexpected background color appears on a gray-scale monitor.

If this parameter is set incorrectly, the monitor may yield unexpected results. Therefore, if a monitor is functioning correctly without it, it is best to omit *:ColorMonitor:* entirely from the configuration file.

### ***:CompensateFloppy:***

CTOS I, CTOS II, and CTOS III

Default: No

Specify **Yes** if the workstation hardware includes a Mode-3 X-Bus module, such as Ethernet or a Voice Processor, which you use concurrently with a floppy disk drive. This prevents read and write errors, which can be caused by concurrent Mode-3 DMA. The compensation that prevents such errors, however, reduces the speed of access to floppy disks.

### ***:ContinueLoadOnError:***

CTOS III

Default: No

Specify **Yes** to allow loading of applications or libraries that cannot be dynamically bound. This is useful in development environments, for example, to load an application when a dynamic library it references does not exist or when a particular reference within a library does not exist. Such references are replaced by zeros at run time, which cause a general protection fault if executed.

### ***:cParExitRunFile:***

CTOS II, CTOS III, and CTOS/XE

Default: 3200

Minimum: 0

Maximum: 65535

This parameter defines the minimum memory partition size for chaining or exiting to another run file. Specify a number of paragraphs.

### ***:cParSpecHeap:***

CTOS II, CTOS III, and CTOS/XE

Default: 128 (768 on CTOS III)

Minimum: 32

Maximum: 4095

This parameter defines the size of the file-specification expansion memory heap. Specify a number of paragraphs.

### ***:cParSysCommonHeap:***

CTOS II, CTOS III, and CTOS/XE

Default: 16

Minimum: 10

Maximum: 4095

This parameter defines the size of the system common heap. Specify a number of paragraphs.

### ***:CrashDumpFile:***

CTOS II and CTOS III

Default: *[Sys]<Sys>CrashDump.sys*

Specify the file specification of the crash dump file to which memory dumps will be written. Note that on the local workstation, this file must be named *CrashDump.sys*, but it may reside on any local disk. To dump memory to the server, the crash dump file must be named *[!Sys]<Sys>Ws>CrashDump.sys*.

Note that this parameter also defines a crash dump file for the extended crash dump utility. For more information, see the *:ExtCrashDumpFile:* parameter, later in this section, and “Collecting a Crash Dump,” in Section 19, “Troubleshooting,” for more information.

### ***:CrashDumpPath:***

CTOS III and CTOS/XE

Default: *[Sys]<Sys>*

This parameter defines a volume and directory for crash dumps and extended crash dumps of processors other than the master processor. (The master processor dumps to *[Sys]<Sys>CrashDump.sys*.) Such crash files are named with the four-digit processor ID, followed by *.crash*, for example, *EP01.crash* (multiprocessor workstation) or *GP01.crash* (SRP).

Specify a volume and directory. On SRPs, the volume you specify must be controlled by the master processor.

### ***:CreateDirectoryProtection:***

CTOS I, CTOS II, CTOS III, and CTOS/XE; processors with disk controllers only

Default: No

This parameter controls whether a volume password is required to create a directory. The default is **No** for CTOS II, CTOS III, and CTOS/XE and **Yes** for CTOS I.

If you specify **Yes**, a volume password must be supplied to create a directory if the volume has a password. If you specify **No**, a password is not required, regardless of whether the volume has a password.



### ***:cSemaphoreRecords:***

CTOS III

Default: 300

Minimum: 0

Maximum: 1024

This parameter defines an amount of memory to reserve for use by semaphores. Specify the number of semaphore records to allocate. Each open semaphore requires one record if unnamed or two records if named. In addition, each process waiting for a noncritical semaphore requires one record. Therefore, specify approximately three times the number of semaphore records that will actually be open simultaneously.

### ***:CursorStart:***

See *:CursorType:*.

### ***:CursorStop:***

See *:CursorType:*.

### ***:CursorType:***

CTOS I, CTOS II, and CTOS III

Default: Underline

This parameter applies to character-mapped workstations (except B24 and B27). Specify **Block** to change the cursor to a block-shaped character. Make the following entries to change the cursor to a double-underline character:

*:CursorType:* Underline

*:CursorStart:* 9

*:CursorStop:* 10

For information about setting a block-shaped cursor on B27-LCW workstations, see the **Screen Setup** command, in the *CTOS Executive Reference Manual*.



**:DebugPort:**

CTOS/XE

Default: No

This parameter configures an asynchronous port for using the Debugger via Basic ATE. This parameter accepts the following values:

**No** specifies that a debugger port is not configured.

**Yes** specifies that the processors's default port is configured as the debugger port using default subparameters. Default ports are as follows:

General Processor	Channel B
Cluster Processor	Channel 3
Terminal Processor	Channel 10

Subparameters, as described below, configure the port with values that you specify.

**:DebugPort:** (Speed = *baud*,  
 Parity = *parity*,  
 Stopbits = *number*,  
 Charbits = *number*,  
 Modem = Yes or No,  
 Port = *alphanumeric*,  
 Processor = *xPnn*)

Subparameters are described below.

Subparameter	Value
Speed	Default: 9600 baud  Specify 50, 75, 110, 134, 150, 200, 300, 600, 1200, 1800, 2400, 4800, 9600, or 19200 baud.
Parity	Default: None  Specify Odd, Even, None, One, Zero, 1 or 0.
Stopbits	Default: 1  Specify 1, or 2.

## Configuring Operating Systems

---

Charbits	Default: 8 Specify 5, 6, 7, or 8.
Modem	Default: No Specify Yes or No.
Port	Default: See below To configure the port of your choice, rather than the default, specify one of the following:  GP <b>A or B</b> (B is the default) GP+CI <b>A to H</b> (B is the default) CP <b>0 to 2</b> (2 is the default) TP <b>0 to 9</b> (9 is the default)
Processor	Default: None  To use the Debugger on a File Processor, Data Processor, or Storage Processor, specify the four-character processor ID of a remote board equipped with an asynchronous port.  When you specify a remote processor, you must also identify a port, as shown in the following example:  <i>:DebugPort: (Processor = GP00, Port = B)</i>

### ***:DiskAllocationLimit:***

CTOS II, CTOS III, and CTOS/XE

Default: All available disk space

Minimum: 512 bytes

Maximum: 4G bytes

This parameter specifies the maximum rate of file expansion, including file creation. It allows other file operations to interleave with disk allocation processing, preventing long periods of file system lockout.

### **:DiskLogThreshold:**

CTOS II, CTOS III, and CTOS/XE; processors with disk controllers only

Default: 0

Minimum: 0

Maximum: 1024

Specify the number of retries that are acceptable before an I/O error is recorded in the system error log. (You can view the system error log with the **PLog** command; see the *CTOS Executive Reference Manual*). By default, every I/O operation requiring a retry is logged. Unrecoverable errors are always logged.

### **:DiskRetryCount:**

CTOS II, CTOS III, and CTOS/XE; processors with disk controllers only

Default: See below

Minimum: 0

Maximum: 1024

Specify the number of retries for a disk operation before it terminates with an I/O error. The default is either 4 or 8, depending on the disk.

### **:ECom:**

CTOS III

Default: Nul

This parameter sets the mode of the ECom port on EISA systems. Specify **RS232**, **TDI**, or **Nul** (async). On Series 4000 and 6000, *[Comm]B* is the ECom port.

**Note:** *The I/O address used for RS232 is 100h through 106h. If a ClientCard is installed in a system using I/O address 100h through 11Fh, the system defaults to asynchronous mode and logs the event.*

**:EISA:**

(Slot = *number*,  
ID = *text*,  
ConfigFile = *FileSpec*)

CTOS III only

Default: See subparameters, below

This parameter identifies an ISA card that does not contain a hardware product ID. EISA cards and some ISA cards contain a product ID that is detected when the system boots. If you install a card that does not contain such an ID, you must assign it an ID with this parameter. Subparameters are described below:

Subparameter	Value
Slot number	Specify the slot number in which the card is installed.
ID	Specify a seven-character ID for the card. In most cases, you will need to know a specific ID that has been assigned by the programmer who wrote the device driver for the card. See the documentation that accompanies the device driver.
ConfigFile	<p>Default: <i>ID.CFG</i></p> <p>Specify the file specification of the <i>.CFG</i> file for the card. <i>.CFG</i> files are written according to EISA specifications and contain detailed hardware device parameters. When you purchase a card, a <i>.CFG</i> should be provided with it. The default is the product ID followed by the suffix <i>.CFG</i>, for example <i>UNB0000.CFG</i>.</p> <p>You can also use this parameter on EISA cards to specify a <i>.CFG</i> file other than the default. (By default, the <i>.CFG</i> for EISA cards matches the hardware product ID and is read automatically when the system boots.) You might do this in situations where you are testing or debugging a card. When you use this parameter for that purpose, you do not need to specify the slot number, since the EISA card is identified directly from its hardware product ID.</p>

### ***:EnterDebuggerOnFault:***

CTOS II, CTOS III, and protected-mode CTOS/XE

Default: No

This parameter applies only when the Debugger is loaded in memory. Specify **Yes** to suspend an application and enter the Debugger when the application cannot recover from a fault. If you specify **No**, applications terminate when they cannot recover from an error. When a system service or the operating system cannot recover, a system crash occurs.

### ***:EVBackgroundOff:***

VAM

Default: Yes

This parameter affects background color on VGA and enhanced-video equipped workstations. Specify **No** to enable enhanced video (EV) background color emulation.

### ***:ExtCrashDumpFile:***

CTOS II and CTOS III

Default: *[Sys]<Sys>CrashDump.sys* (if large enough)

**Note:** *Extended crash dumping is not supported on EISA/ISA workstations.*

This parameter defines a file for extended crash dumping when auto dumping is enabled (see *:SuppressAutoDump:*) and if *[Sys]<Sys>CrashDump.sys* is not large enough to contain the entire dump.

Make the extended crash dump file large enough to contain the entire contents of memory. To determine the correct size, multiply the total amount of memory, in K bytes, by 2. For example, if the workstation has 2048K bytes of memory, allocate 4096 sectors when you create the extended crash dump file.

An extended crash dump file is not created when the disk is initialized; you must create it with the **Create File** command (see the *CTOS Executive Reference Manual*).



### ***:ExtCrashVDMFile:***

CTOS II and CTOS III

Default: *[Sys]<Sys>Vdm\_Dmy.run*

**Note:** *Extended crash dumping is not supported on EISA/ISA workstations.*

This parameter specifies the video run file to be used when the system is performing an extended crash dump. It must be in the form of a full file specification. The default is a small video program that does not update the video hardware and, therefore, does not display the extended crash dump operation while it is executing.

In many cases, the video run file *Vdm\_Ch.run* can be specified; it is small enough to be loaded during crash dumps and allows the crash dump operation to be displayed. In most cases, unless the workstation is loading a very small operating system, the video files *Vdm\_VGA.run* and *Vdm\_BM.run* are too large to be loaded during extended crash dump operations.

### ***:fAllowCommLineDMAOnCPU:***

CTOS II and CTOS III

Default: No

This parameter affects the serial ports on B39 (Series 386i) workstations. Specify **Yes** to initialize DMA for both channel A and channel B. If you specify **No** or leave this field blank, DMA is not initialized.

### ***:FileCacheDefault:***

(Enable = *Yes/No*,  
WriteBehind = *Yes/No*,  
CleanPriority = *number*)

CTOS III (for CTOS II and CTOS/XE, use *:FileCacheDefaultEnable:*, described below)

Default: See subparameters, below

This parameter sets default conditions for file system caching; however, it alone does not configure or implement file system caching on the processor; see the *:FileCacheService:* parameter, below. See also Section 14, "Using a File-System Cache," for detailed information about file cache configuration requirements.



Subparameters are described below.

<b>Subparameter</b>	<b>Value</b>
Enable	<p>Default: No</p> <p>Specify <b>Yes</b> to enable caching on all files that are not specifically disabled. Specify <b>No</b> to disable caching on all files that are not specifically enabled.</p>
WriteBehind	<p>Default: No</p> <p>Specify <b>Yes</b> to enable write-behind caching, where changes to cached sectors are written to disk only at prespecified intervals. See Section 14, "Using a File System Cache," for more detailed information about write-behind caching.</p>
CleanPriority	<p>Default 128</p> <p>If you specified Yes to the WriteBehind subparameter (above), specify a number from 1 to 255 that represents the process priority for write-to-disk-from-cache operations. The default value of 128 is a priority that is lower than foreground application processes.</p>

***:FileCacheDefaultEnable:***

CTOS II and CTOS/XE (for CTOS III, use *:FileCacheDefault:*, described above)

Default: Yes

This parameter sets the default condition for file system caching. Specify **Yes** to enable caching on all files that are not specifically disabled. Specify **No** to disable caching on all files that are not specifically enabled.

This parameter alone does not configure or implement file system caching on the processor; see the *:FileCacheService:* parameter, below. See also Section 14, "Using a File-System Cache," for detailed information about file cache configuration requirements.

**:FileCacheService:** (BlockSize = *bytes*,  
BlockCount = *number*,  
MinWorkingSetBlockCount = *number*  
DirtyBlockCount = *number*)

CTOS II, CTOS III, and protected-mode CTOS/XE

Default: See subparameters, below

This parameter allots a file-system cache. On multiprocessor workstations, the default value applies to the master processor only. On SRPs, the default value applies to GP+SI boards only. Subparameters are described below.

Subparameter	Value
BlockSize	Default: 4096  Specify the number of bytes to be allocated for each block. It must be a multiple of 4096.
BlockCount	Default: See below Minimum: 64K bytes divided by block size Maximum: Depends on available memory  Specify the number of blocks to be allocated. Block count multiplied by block size equals the total amount of memory required for the cache. In most cases, increase this value, rather than <i>BlockSize</i> , to increase the size of the cache. Default values are 128 on SRP GP+SI board and on workstation servers; including IOPs. On other SRP boards and on client workstations, the default value is 0.
MinWorkingSetBlockCount	Default: See below  Specify the number of blocks that cannot contain locked-in files. The default value is 64K bytes divided by the block size. For example, if block size is 4096 bytes, the default value is 16. If you change the block size, the default value changes accordingly.
DirtyBlockCount	Default: 10 Minimum: 1 Maximum: 100  Specify the number of cache blocks that may be dirty (written to) before data in the cache is written back to disk. Any write-to-cache operation that exceeds that percent forces cache blocks to be written immediately to disk.

### ***:FileStructureVerify:***

CTOS II, CTOS III, and CTOS/XE; processors with disk controllers only

Default: No

This parameter is used for diagnosing file system or hardware problems that could result in corrupted disks. If you specify **Yes**, file headers, directory entries, bit-map pages, and volume home blocks are verified for data integrity immediately after they are written to disk. That verification, however, degrades the speed of file system performance. If you specify **No**, a verification is not performed.

### ***:ForceBlackBackground:***

VAM; VGA only

Default: No

This parameter applies to gray-scale monitors only; it has no effect on color monitors. Specify **Yes** to set the background color to black. If this parameter is set to **No** or left blank, the background color is white.

### Caution

The following parameter, *:ISOVideoCompliant:*, should be used only with monitors capable of driving a refresh rate of greater than 75Hz. (See the documentation that accompanies your monitor). Enabling this parameter on monitors that are not capable of the required refresh rate could cause damage to the monitor.

### *:ISOVideoCompliant:*

VAM; VGA only

Default: No

This parameter enables the ISO-compliant refresh rate of 75Hz or greater on the CTOS VideoCard, B25VGx, and GC-x04 graphics controllers. (ISO stands for International Standards Organization.) The CTOS VideoCard supports ISO compliance in the following modes:

Columns	Rows
80	29
80	34
80	38
132	29
132	34
146	38

The B25VGx and GC-x04 graphics controllers support ISO compliance in the 80 by 29, 80 by 34, 132 by 29, and 132 by 34 modes only.

### *:KbdProfile:*

(ID = *KbdID*,  
AltNlsStyle = *Yes/No*)

CTOS II and CTOS III

Default: See subparameters, below

This parameter defines the keyboard profile to be loaded at boot time. Subparameters are described below.

Subparameter	Value
ID	<p>Default: 04h</p> <p>Specify the keyboard profile ID for the workstation. The default is 04h unless NLS Table 15 is present in <i>Nls.sys</i>, in which case the default is B0h. See the <i>CTOS Operating System Concepts Manual</i> for information about keyboard profiles.</p>
AltNlsStyle	<p>Default: No</p> <p>This subparameter applies only to workstations that are configured with <i>Nls.sys</i>. Specify <b>Yes</b> to override the default keyboard style the operating system reads from <i>Nls.sys</i>. The recognized keyboard style is taken from NLS Table 15 if it is present; otherwise, it is taken from NLS Table 0.</p>

***:KbdTables:*** (Number = *number*, Size = *bytes*)

CTOS II and CTOS III

Default: See subparameters, below

This parameter defines the amount of memory required for loadable keyboard data blocks and the maximum number of them to be loaded. The defaults (described below) allow for one translation data block and one emulation data block to be loaded. Additional data blocks may be needed by certain applications, either because the application itself expects to load a particular keyboard table or the run file header has been set to expect one. Subparameters are described below.

Subparameter	Value
Number	<p>Default: 1</p> <p>Specify the number of translation data blocks and/or emulation data blocks that will be required.</p>
Size	<p>Default: See below</p> <p>Specify the maximum amount of memory, in bytes, required for loadable keyboard data blocks. The default value is the number of bytes required for the largest translation data block and the largest emulation data block in <i>NlsKbd.sys</i>.</p>



### ***:LibrarySearchPath:***

CTOS III

Default: *[Sys]<PMDLL>, [Sys]<Sys>*

This parameter defines a list of paths to be searched for dynamic link libraries. Each entry consists of a volume, directory, and optionally, a password. You may specify more than one path. Surround path entries with parentheses and separate them with commas. A period (.) denotes the user's current path. The following example illustrates multiple search-path entries:

```
:LibSearchPath: ([Sys]<Sys>, [Sys]<Lib>^pswd, .)
```

### ***:LfsToMaster:***

CTOS I, CTOS II, and CTOS III; client workstations with local file systems (LFS) only

Default: No

This parameter instructs the operating system to search the specified paths or file specifications on the server's system volume (in addition to those on the LFS) for read-only files. To enable all paths, specify **All**.

To enable selected multiple paths, use multiple entries of this parameter, as shown in the following example:

```
:LFSToMaster:[Sys]<Sys>*  
:LfsToMaster:[Sys]<GPS>*
```

If you specify a wild card (\*), it must be the last character on the line, for example:

```
:LfsToMaster:[Sys]<Sys>FileName*
```

Embedded wild cards are not allowed. If no wild card is present, only the specified file name is enabled, for example:

```
:LfsToMaster:[Sys]<Sys>FileName.msg
```

When this option is enabled, it can be suppressed by specifying **[+Sys]** in a file specification. That instructs the operating system to search the local path only.



The parameter values **Yes** and **No** are also supported. **Yes** is equivalent to `:LfsToMaster:[Sys]<Sys>*`, and **No** disables the option, if it is the last instance of this entry in the configuration file.

### ***:LoadableRequestFile:***

CTOS III and CTOS/XE

Default: `[Sys]<Sys>Request.sys`

This parameter defines the file specification of the request file to be loaded when a processor boots. It must be located on a volume connected to the master processor.

### ***:LoadDebugger:***

CTOS III and CTOS/XE

Default: No

Specify **Yes** to load the Debugger into memory when the workstation boots. If you specify **No** or leave this field blank, the Debugger is not loaded.

### ***:LogUnknownEntries:***

CTOS III and CTOS/XE

Default: Yes

This parameter determines whether occurrences of unrecognized entries in the operating system configuration file are written to the system log file. It is placed first in the file, before the `:Boot:` list and `:Processor:` entries.

If you specify **No**, unrecognized entries are not logged. If you specify **Yes**, unrecognized entries are written to the system log file, which you can view or print with the **PLog** command.

### ***:MapKeyboardID:*** (Source = *KbdID*, Target = *KbdID*)

CTOS I, CTOS II, and CTOS III; (see the note for CTOS I, below)

Default: See subparameters, below

This parameter controls how various styles of keyboards are mapped to available keyboard data blocks. When the keyboard hardware matches the ID specified as the *Source* subparameter, it is recognized as the keyboard specified as the *Target* subparameter and keyboard data blocks are loaded accordingly.

## Configuring Operating Systems

---

Keyboard IDs are specified as hexadecimal numbers and must be followed by the character **h**. In addition, when the hexadecimal entry begins with an alpha character, the alpha character must be preceded by the numeral **0**, for example, 0E1h.

This parameter can appear multiple times within the configuration file. Subparameters are described below.

Subparameter	Value
Source	Enter the hexadecimal keyboard ID for the keyboard you want to use.
Target	Enter the hexadecimal keyboard ID for the keyboard to which you want to map.

**Note:** *CTOS I (real-mode) operating systems do not recognize subparameters. Therefore, the format of this entry consists of two 2-digit hexadecimal values, such as 040B, where the first two digits identify the source keyboard and the last two digits identify the target.*

**:MassStorage:** (see below for subparameters)

CTOS II, CTOS III, CTOS/XE

Default: None

This parameter defines device names and device passwords for disk drives on SRPs. It is also used to configure memory disks on both workstations and SRPs.

### Subparameters for SCSI Drives:

**:MassStorage:** (Class = *type*,  
Adaptor = *number*,  
Target = *number*,  
LUN = *number*,  
Device = *name*,  
Password = *password*)

Subparameters are described below. See Figure 16-3 for an example.

Subparameter	Value
Class	Default: SCSI Specify <b>SCSI</b> .
Adaptor	Default: 0 Specify <b>0</b> for devices connected to the first channel on the SCSI interface, <b>1</b> for the second channel.
Target	Default: 0 Specify a number from <b>0</b> to <b>7</b> to designate the target setting on the device (see the documentation for your SCSI device).
LUN	Default: 0 Specify <b>0</b> . (At a future time, logical unit numbers from 0 to 63 will be supported.)
Device	Default: See below Specify a device name to be assigned to the drive. By convention, SCSI disk devices are named <i>dn</i> , where <i>n</i> is a number (usually matching the <i>Target</i> subparameter setting), for example <i>d1</i> . Default device names for SCSI drives connected to the master processor are <i>d1</i> to <i>d15</i> . If, however, fewer than fifteen SCSI disks are connected to the master processor, you can reassign device names to disks controlled by different processors. For example, if the master processor controls <i>d1</i> to <i>d3</i> , you can continue sequential numbering ( <i>d4</i> to <i>d7</i> ) in the next cabinet.
Password	Default: See below Specify a password to be assigned to the drive. By convention, device passwords match device names. For disks connected to the master processor, default device passwords match device names.

### Subparameters for Non-SCSI Drives:

*:MassStorage:* (Class = *type*,  
Unit = *number*,  
Device = *name*,  
Password = *password*)

Subparameters are described below. See Figure 16-3 for an example.

Subparameter	Value
Class	Specify <b>ST506</b> or <b>SMD</b> .
Unit	Specify a number to denote the position of the drive's connection to the processor (see the appropriate hardware installation manual). For a memory disk, use the number you assign in the device name (see the <i>Device</i> subparameter, below).
Device	Default: See below  Specify a device name to be assigned to the drive. By convention, ST-506 disk drives are named <i>dn</i> , where <i>n</i> is a number (for example, <i>d1</i> ); SMD drives are named <i>sn</i> (for example, <i>s1</i> ). Default device names for the master processor are <i>d1</i> to <i>d3</i> for ST-506 drives and <i>s0</i> to <i>s5</i> for SMD drives.
Password	Default: See below  Specify a password to be assigned to the drive. For disks connected to the master processor, default device passwords match device names.

**Subparameters for Memory Disks:**

```
:MassStorage:          (Class = type,
                          Unit = number,
                          Device = name,
                          Password = password
                          Volume = name
                          MaxSectors = number
                          MaxDirectories = number
                          MaxSysFiles = number
                          MaxTempFiles = number
                          MaxFiles = number)
```

Subparameters are described below. See Figure 16-3 for an example.

Subparameter	Value
Class	Specify <b>CacheMemory</b> or <b>Memory</b> .  A cache memory disk can be used as a scratch volume; see "Using a Cache Memory Disk" in Section 15, "Optimizing System Performance." A memory disk can also be used to create a bootable tape for software installation; see the <b>Create Boot Tape</b> command in the <i>CTOS Executive Reference Manual</i> .
Unit	Specify a number to uniquely identify each memory disk or cache memory disk on the processor. In most cases, this value will be 0, because it is unusual to configure more than one disk in memory per processor.
Device	Specify a device name. You can assign any name, but by convention, cache memory disks are named <i>CMdn</i> (for example, <i>CMd0</i> ) and memory disks are named <i>mn</i> (for example, <i>m0</i> ), where <i>n</i> matches the number you assigned as the Unit subparameter.
Password	Specify a password to assign to the memory disk.
Volume	Specify a volume name for the cache memory disk.
MaxSectors	Default: 128 Minimum: 1 Maximum: Based on available memory  Specify the maximum amount of memory, in sectors, to allot to the cache memory disk.

## Configuring Operating Systems

---

MaxDirectories	Default: 1 Minimum: 1 Maximum: Based on size of memory disk  Specify the maximum number of directories that can be created on the cache memory disk.
MaxSysFiles	Default: 1 Minimum: 1 Maximum: Based on size of memory disk  Specify the maximum number of files that can be created in the cache memory disk's <Sys> directory.
MaxTempFiles	Default: 4 Minimum: 4 Maximum: Based on size of memory disk  Specify the maximum number of files for the <\$000> directory.
MaxFiles	Default: 15 Minimum: 15 Maximum: Based on size of memory disk  Specify the maximum number of files that can be created on the cache memory disk.

### ***:MaxConcurrentQuiet:***

CTOS II and CTOS III

Default: 30 on servers, 5 on client workstations

This parameter controls the number of “quiets” that can occur simultaneously. A “quiet” is the act of notifying server programs of the termination of another program. The terminated program may be local or remote. If this parameter is exceeded during operation, the system holds additional quiets until outstanding quiets are complete. Each quiet unit consumes approximately 30 bytes of memory. Too small a number can impair performance.



### ***:MaxConcurrentTerm:***

CTOS II and CTOS III

Default: 15

This parameter controls the maximum number of program terminations that can occur simultaneously. If this number is exceeded during operation, the system will crash with Error 820 (termination heap full). Each termination unit consumes approximately 144 bytes of memory.

### ***:MaxHugeSg:***

CTOS III

Default: 128

Minimum: 0

Maximum: 1024

This parameter defines the maximum number of contiguous GDT selectors that are allocated to the system and available to `AllocHugeMemory` (see the *CTOS Procedural Interface Reference Manual*).

### ***:MaxInstanceSeg:***

CTOS III

Default: 2048

Minimum: 0

Maximum: 5120

This parameter defines the maximum number of instance selectors allocated to the system. Specify a number of bytes.

### ***:MemoryScrub:***

CTOS III

Default: No

Specify **Yes** if you want the operating system to preset all allocated memory to zero. If you add `:MemoryScrub:Yes` to your configuration file, the paging service has to write pages of memory to the swap file more frequently than if it is set to **No**.

### Caution

Do not change the value of the *:MaxXBlocksOut:* parameter, described below, unless instructed to do so by a Technical Support engineer. An incorrect value can cause total system failure.

#### ***:MaxXBlocksOut:***

CTOS II, CTOS III, and CTOS/XE; processors with cluster lines only

Default: 5

Minimum: 1

Maximum: Number of allocated X-blocks

Specify the maximum number of X-blocks that can be outstanding for each workstation.

#### ***:Mode3DMAMaster:***

See *:ModuleType:*.

#### ***:ModuleType:***

#### ***:XBusWindowSize:***

CTOS I, CTOS II, and CTOS III

Default: None

When a workstation bootstraps, the operating system configures the X-Bus window for each module it identifies (see "X-Bus Management," in the *CTOS Operating System Concepts Manual*). If you are using a nonstandard module, or if the documentation so instructs you, add these two parameters to the *Config.sys* file. Obtain the values from the documentation for the module.

If you specify a *:ModuleType:* parameter, it must be immediately followed by an *:XBusWindowSize:* parameter, as shown above.

In addition, the *:Mode3DMAMaster:* and *:UsedFromVirtualRealMode:* parameters are used only in conjunction with the *:ModuleType:* and *:XBusWindowSize:* parameters on CTOS II and CTOS III only, as described below.

The *:Mode3DMAMaster:* parameter follows a *:ModuleType:* parameter. Specify **Yes** to indicate that the workstation is equipped with a Mode-3 DMA master module. When set to **Yes**, this parameter controls the loading of certain protected-mode programs. Those that have not been explicitly linked to load above 16M bytes of memory will be loaded somewhere within the first 16M bytes. (See the *CTOS Programming Utilities Reference Manual: Building Applications*, for information about linking programs.)

The *:UsedFromVirtualRealMode:* parameter is used when the workstation is equipped with a nonstandard X-Bus module. It must be placed immediately after the *:ModuleType:* and *:XBusWindowSize:* parameters. Specify **Yes** if a real-mode program will be making calls to the MapXBusWindow operation (see the *CTOS Procedural Interface Reference Manual*). If you specify **No** or omit this field, Error 38 (invalid window size) occurs when a real-mode program attempts to call MapXBusWindow for the nonstandard module.

### ***:nMaxNetUsers:***

CTOS II and CTOS III; servers only

Default: See below

Minimum: See below

Maximum: 1024

This parameter defines the number of incoming requests from remote BNet nodes that can be served simultaneously on this workstation.

On protected mode servers, default values are 20, 40, and 60 for small, medium, and large operating systems, respectively. On virtual memory servers, the default value is 40. Minimum values are the same as the default values.

### ***:NoBeep:***

CTOS III

Default: No

This parameter defines the status of the audible system alarm. By default, it is enabled. Specify **Yes** to disable it.

### **Caution**

Do not change the value of the *:nRepollActive:* parameter, described below, unless instructed to do so by a Technical Support engineer. An incorrect value can cause total system failure.

### ***:nRepollActive:***

CTOS/XE; processors with cluster lines only

Default: 0

Minimum: 0

Maximum: 5

Specify the number of times to repoll workstations.

### ***:nRkusUsers:***

CTOS III and CTOS/XE

Default: 2

Minimum: 0

Maximum: 14

This parameter defines the number of Remote User Manager sessions that can be started on a protected-mode processor. Specify a number. Although you can specify a large number of sessions, the maximum number of users is actually determined by the Remote User Manager.

### ***:OldMaster:***

CTOS I, CTOS II, and CTOS III

Default: No

This parameter accelerates booting for workstations that boot from a real-mode server. If the server is not running a protected-mode operating system, specify **Yes**.

### ***:OSNameClstr:***

### ***:OSNameClstrLfs:***

### ***:OSNameSrvr:***

### ***:OSNameStnd:***

CTOS III

Default: vClstr, vClstrLfs, vSrvr, and vStnd, respectively

These parameters change the operating system version string that is displayed by the Executive for each type of operating system, respectively. The maximum size of the string is 20 bytes, and excess bytes are truncated. The version string that you specify is followed by the official version number of the CTOS III operating system, for example:

CTOS III Workstation 1.1

This parameter does not affect the version string in the operating system image file, and the original version string is also displayed by the Debugger and the **Version** command.

### ***:PagingServiceFileCount:***

CTOS III

Default: 100

This parameter defines the maximum number of files that can be handled by the paging service. It is provided so that the number of file information blocks (FIBs) can be increased if Error 13969 (too many backing store files) occurs. One FIB must be allocated for each file. Specify the number of FIBs to allocate.

### ***:PagingServiceHeapSize:***

CTOS III

Default: See below

This parameter defines the size of the paging service heap. It is provided so that the heap size can be increased if Error 13963 (heap memory not available) occurs. The default value is the number of paging service users times the size of the paging service stack. Specify a number of bytes.

### ***:PagingServiceMaxWritesInProgress:***

CTOS III

Default: 20

This parameter defines the maximum number of write operations that may be in progress simultaneously. It is used to limit the number of outstanding page writes to the swap file.

### ***:PagingServicePrefaultCount:***

CTOS III

Default: 2

Minimum: 0

Maximum: 14

This parameter defines the number of pages the paging services attempts to bring into memory in anticipation of accessing them. If it is set to 0, the paging service does not attempt to prefault any pages.

### ***:PagingServicePrefaultCountMax:***

CTOS III

Default: 14

Minimum: 0

Maximum: 14

This parameter defines the maximum number of pages that can be prefaulted if the *:PagingServicePrefaultCount:* parameter is enabled.



***:PagingServiceStackSize:***

CTOS III

Default: 500

This parameter defines the size of the paging service stack. It is provided so that the stack size can be increased if Error 13962 (stack overflow) occurs. Specify a number of bytes.

***:Processor:***

CTOS III and CTOS/XE

See "Processor Section," earlier in this section.

***:RebootClusterOnMasterDown:***

CTOS II and CTOS III; client workstations only

Default: None

This parameter defines one or more hardware ID numbers, which, if matched by the hardware ID number assigned to the client workstation, causes that client workstation to be rebooted following a loss of communication with the server. Specify a hardware ID number from 1 to 127. This parameter may be repeated for multiple hardware ID numbers. It must be included in *WsNNN>Config.sys* or *Ws>Config.sys* if one of those files is being used instead of *Config.sys* for client workstations that boot from the server.

Hardware IDs are assigned with the **Write Hardware ID** command; see the *CTOS Executive Reference Manual*. In addition, hardware IDs are not supported on all workstations. See the documentation for your processor model.

***:RemoteCacheClient:*** (Name = *client*,  
Pool = *pool*)

Protected mode CTOS/XE

Default: None

This parameter defines a real-mode processor that can access a remote cache. Multiple occurrences of this parameter are permitted so that more than one remote processor can be defined.

Subparameters are described below.

Subparameter	Value
Name	Specify the four-character ID of the processor that is to access a remote cache.
Pool	Specify the name of the cache pool the processor is to access. (The cache pool must be defined with the <i>:RemoteCachePool:</i> parameter.)

***:RemoteCachePool:*** (Name = *pool*,  
Password = *password*,  
BlockSize = *bytes*,  
BlockCount = *number*,  
MinWorkingSetBlockCount = *number*)

Protected mode CTOS/XE

Default: None

This parameter defines a remote cache pool that can be accessed by real-mode processors. Multiple occurrences of this parameter are permitted to define more than one cache pool.

Subparameters are described below.

Subparameter	Value
Name	Specify the name of a cache pool to be used by remote processors. Specify <b>Local</b> to enable remote access of the cache installed with the <i>:FileCacheService:</i> parameter. Or, define an additional cache by specifying a unique name that does not conflict with any device name on the system. Suggested names are <i>c0</i> , <i>c1</i> , and so on.
Password	Default: No password  Specify a password if you want to assign one to the remote cache pool.
BlockSize	Default: 0  (See the <i>:FileCacheService:</i> subparameters for minimum and maximum values.) If you specify a unique cache-pool name, specify the size of blocks to be allocated for the remote cache. If you specify <b>Local</b> as the cache pool name, leave this field blank.

BlockCount

Default: 0

(See the *:FileCacheService:* subparameters for minimum and maximum values.) If you specify a unique cache pool name, specify the number of blocks to be allocated for the remote cache. If you specify Local as the cache pool name, leave this field blank.

MinWorkingSetBlockCount

Default: 0

(See the *:FileCacheService:* subparameters for minimum and maximum values.) If you specify a unique cache-pool name, specify the number of blocks that cannot contain locked-in files. If you specify "Local" as the cache pool name, leave this field blank.

***:RemoteCacheService:***

(Priority = *number*,  
StackSize = *bytes*,  
Descriptors = *number*)

Protected mode CTOS/XE

Default: See subparameters, below

This parameter establishes optional internal parameters for the remote cache service. Subparameters are described below.

**Caution**

Do not change the value of the *Priority*, *StackSize*, or *Descriptors* subparameters, described below, unless instructed to do so by a Technical Support engineer. An incorrect value can cause total system failure.

**Subparameter**

**Value**

Priority

Default: 7  
Minimum: 0  
Maximum: 127

Specify a value from **0** (highest priority) to **127** (lowest priority) for remote cache requests.

StackSize

Default: 512 bytes  
Minimum: 512 bytes  
Maximum: No maximum

Specify the number of bytes for the remote-cache stack size.

Descriptors

Default: 100

Minimum: 33

Maximum: 1000 (approximately)

Specify the number of descriptors for the remote cache. These determine the number of concurrent requests the remote cache service can process.

### ***:RepeatKeyFactor:***

CTOS I, CTOS II, and CTOS III

Default: 0

This parameter controls the rate at which characters are repeated while a key remains depressed in applications that read the keyboard in character mode. Enter one of the following values:

- 0** Normal (default) speed
- 1** Half speed
- 2** One-fourth speed
- 3** No repeating

### ***:RequestTracker:***

CTOS/XE

Default: No

This parameter enables request tracking. Specify **Yes** if you want outstanding requests to be copied to an exchange for tracking. (See the *:RqTracker:* parameter for workstations.)

### ***:RkvsFile:***

CTOS III and CTOS/XE

Default: *[Sys]<Sys>Rkvs.run*

This parameter defines the name of the Remote Keyboard Video Service run file to be loaded when the processors boots.

Specify the file specification of the RKVS run file. It must be located on a volume attached to the master processor.

### **:RqTracker:**

CTOS II and CTOS III

Default: No

This parameter enables request tracking. Specify **Yes** if you want outstanding requests copied to an exchange for tracking. (See the *:RequestTracker:* parameter for SRPs.)

### **:sBroadcastHeap:**

CTOS III and CTOS/XE

Default: 1024

Minimum: 100

Maximum: 65520

This parameter defines the count of bytes to limit the number of outstanding requests being broadcast. Specify a number of bytes.

### **:ScratchVolumeName:**

CTOS II, CTOS III, and CTOS/XE

Default: *Sys*

This parameter specifies the disk that is used as the “scratch” volume (*[Scr]*). The scratch volume is used for temporary storage by certain commands, such as **Floppy Copy**. Scratch files can consume a lot of disk space, and if enough disk space is not available, the command can fail. Therefore, you may want to specify the scratch volume to be a disk with a large amount of free disk space; such a disk is frequently not *[Sys]*.

Specify a volume or device name, with or without square brackets, for example, *[BigDaddy]* or *d1*. If you specify a volume other than *[Sys]* as the scratch volume, you must create a directory named *<\$000>*, with a capacity of 750 files, on that disk.

On SRPs, the disk you specify must be connected to the master processor.

On workstations booted from a local disk, the scratch volume must also be a local disk. On workstations booted from the server, the scratch volume may be either a local disk or the system volume on the server.



### ***:ScreenTimeout:***

CTOS II and CTOS III

Default: 0 (see below)

Minimum: 0

Maximum: 109

This parameter controls the amount of time that elapses before the screen is shut off when the workstation is not being used. Specify a number of minutes. The default, 0, means that the screen always stays on.

### ***:SCSIManagerName:***

CTOS/XE; processors with disk controllers only

Default: See below

Enter up to 12 characters to specify a SCSI manager name for the processor. This name is used by applications to route requests to multiple SCSI managers.

Default values begin with SCSI, for the SCSI manager on the first GP, SCSI1, for the second, and so on.

### ***:sEnvironment:***

CTOS II and CTOS III

Default: 1024

Minimum: 0

Maximum: 65536

This parameter defines an amount of environment memory for each user. It is provided so that the amount of environment memory can be increased if an error occurs, indicating that the environment is full. Specify a number of bytes.



**:SequentialStorage:** (see below for subparameters)

CTOS III and CTOS/XE; processors with tape drive controllers only

Default: None

This parameter defines the device name for a tape drive. On an SRP, this parameter can also be implemented on the QIC Interface board (which controls non-SCSI QIC tape drives) by making an entry for any one of the processors in the same cabinet. For clarity in the configuration file, such an entry is usually listed under the File Processor or General Processor adjacent to QIC Interface board.

**Subparameters for SCSI Drives:**

**:SequentialStorage:** (Class = *type*,  
 Adaptor = *number*,  
 Target = *number*,  
 LUN = *number*,  
 Device = *name*,  
 Password = *password*)

Subparameters are described below. See Figure 16-3 for an example.

Subparameter	Value
Class	Default: None Specify <b>SCSI</b> .
Adaptor	Default: 0 Specify <b>0</b> for devices connected to the upper channel on the SCSI interface, <b>1</b> for the lower channel.
Target	Default: 0 Specify a number from <b>0</b> to <b>7</b> to designate the target setting on the device (see the documentation for your SCSI device).
LUN	Default is 0. Specify <b>0</b> .

Device	Default: None  Specify a device name to assign to the tape drive, including square brackets. This can be any name you choose, for example, <i>[QIC]</i> , <i>[QIC0]</i> , or <i>[Fred]</i> .
Password	Default: None  Specify a password to assign to the drive. By convention, device passwords match device names.

### Subparameters for Non-SCSI Tape Drives:

*:SequentialStorage:* (Class = *type*,  
Unit = *number*,  
Device = *name*,  
Password = *password*)

Subparameters are described below. See Figure 16-3 for an example.

Subparameter	Value
Class	Default: None  Specify <b>QIC36</b> or <b>Halfinch</b> .
Unit	Default: None  This subparameter applies to half-inch tape drives only. Specify the number between 0 and 7 that corresponds to the ID switch setting on the tape drive unit. (See the documentation for the tape drive.)
Device	Default: None  Specify a device name to assign to the tape drive, including square brackets. This can be any name you choose, for example, <i>[QIC]</i> , <i>[QIC0]</i> , or <i>[Dan]</i> .
Password	Default: None  Specify a password to assign to the drive. By convention, device passwords match device names.

***:sExitProcStack:***

CTOS III

Default: 2048

Minimum: 0

Maximum: 65536

This parameter defines the size of the stack created for ExitList procedures. Specify a number of bytes. (See SetExitList in the *CTOS Procedural Interface Reference Manual*. See also the section about dynamic link libraries in the *CTOS Operating System Concepts Manual*.)

***:sLoaderHeap:***

CTOS III

Default: 24576

Minimum: 0

Maximum: 65536

This parameter defines the size of the loader's heap. It is provided so that the size of this heap size can be increased if Error 1114 (loader heap full) occurs. Specify a number of bytes.

***:sMemMgrHeap:***

CTOS III

Default: 16384

Minimum: 0

Maximum: 65536

This parameter defines the size of the memory manager's heap. It is provided so that the size of this heap can be increased if Error 477 (memory manager heap full) occurs. Specify a number of bytes.

### **:sNameHash:**

CTOS III

Default: 32

Minimum: 1

Maximum: 512

This parameter defines the number of name table positions in a class. For detailed information, see "Managing Names," in the utility operations section of the *CTOS Operating System Concepts Manual*.

### **:sNameHeap:**

CTOS III

Default: 61440

Minimum: 0

Maximum: 1048576

This parameter defines the size of the name heap. Specify a number of bytes.

### **:SrvrAgentFile:**

CTOS III

Default: See below

This parameter specifies the name of the file that contains the loadable LAN software for CTOS III servers. It is used to convert a CTOS III standalone operating system into a server operating system for single-processor workstations. It does so by loading the cluster software after the standalone operating system has booted. That conserves memory on single-processor workstations, which do not require the multiprocessor capabilities of the prepackaged server operating system.

The default for the CTOS III server operating system is *[Sys]<Sys>Srvr.sys*, which automatically loads the cluster software. The default for the CTOS III standalone operating system is None. Therefore, in cases where a true standalone system is wanted, simply omit this parameter.

This parameter can also be used to load a different LAN protocol from the standard CTOS cluster. Specify the file specification of the file containing the LAN software you want to use. See also *:ClientAgentFile:*, earlier in this section.

### ***:SuppressAutoDump:***

CTOS II and CTOS III

Default: No

This parameter implements automatic extended crash dumping. On systems that require it, an extended crash dump is performed after the first megabyte of memory has been dumped. If the crash file, *<Sys>CrashDump.sys*, is large enough to contain the extended crash dump, it will be used. If not, the file specified for *:ExtCrashDumpFile:* is used (see that parameter, above).

If you specify **Yes**, an extended crash dump does not take place. You can, however, perform it manually with the **Extended Crash Dump** command; see the *CTOS Executive Reference Manual*.

### ***:SuppressCommDMABufferCheck:***

CTOS III

Default: No

This parameter determines whether communications buffers are checked for contiguity before DMA is started. By default, checking is enabled. Specify **Yes** to disable checking.

### ***:SuppressDebugger:***

CTOS II and CTOS III

Default: No

This parameter controls whether the Debugger is loaded into memory when the workstation boots. It applies only if the Debugger software has been installed and is present in *[Sys]<Sys>*. By default, the Debugger is loaded. If you specify **Yes**, the Debugger is not loaded.

### ***:SuppressGlobalPolicy:***

CTOS III

Default: No

This parameter controls the global paging policy, which allows applications to compete for 4K-byte pages of memory. Specify **Yes** to suppress global paging and limit the number of pages consumed by an application to a number derived from the partition size that is specified in the Context Manager configuration file. See also “Optimizing Virtual Memory,” in Section 15, “Optimizing System Performance.”

### ***:SuppressPmDefaults:***

CTOS III

Default: No

This parameter is useful for quickly configuring a system for use without Presentation Manager. It decreases various system resources to reflect the absence of Presentation Manager and saves approximately 135K bytes of memory.

Specify **Yes** to set the following configuration parameters:

- :cSemaphoreRecords: 8*
- :MaxInstanceSeg: 0*
- :sNameHeap: 0*
- :sExitProcStack: 512*
- :sMemMgrHeap: 512*
- :sLoaderHeap: 4096*
- :sEnvironment: 0*

(Each of the parameters listed above is also described individually, in alphabetical order, throughout this section.)

Parameters in *Config.sys* are read from top to bottom, and when a parameter occurs more than once, the last occurrence in the file is the one that remains set. Therefore, to override a parameter that is set by *:SuppressPmDefaults:*, the keyword for the parameter you want to override must be placed after the *:SuppressPmDefaults:* keyword in *Config.sys*.



In the following example, the system is configured for 100 semaphore records, because the value specified for *:cSemaphoreRecords:* overrides the value of 8 that was set by *:SuppressPmDefaults:*.

```
:SuppressPmDefaults: Yes  
:cSemaphoreRecords: 100
```

In the following example, however, the system is configured for 8 semaphore records, because the value of 8 that is set by *:SuppressPmDefaults:* overrides the value that was specified for *:cSemaphoreRecords:*.

```
:cSemaphoreRecords: 100  
:SuppressPmDefaults: Yes
```

### ***:SwapFile:***

CTOS II, CTOS III, CTOS/XE

Default: *[Sys]<Sys>CrashDump.sys*

This parameter specifies the file to which the contents of a memory partition are swapped. If the specified file is not available (as in the case of a diskless workstation) or if it is too small and cannot be expanded (as in the case of *CrashDump.sys*) the alternate swap file is used. See the *:SwapFileAlternate:* and *:SwapFileSize:* parameters, below.

If the swap file you specify requires a password, a valid password for it must be built (SysGen'd) into a customized version of the operating system. See Section 17, "Building a Customized Operating System."

### ***:SwapFileAlternate:***

CTOS II and CTOS III

Default: *[Sys]<Sys>SwapAreaNNN.sys*

This parameter sets the swap file specification to be used when the primary swap file, as specified in the *:SwapFile:* parameter, cannot be accessed. The default specification is *[Sys]<Sys>SwapAreaNNN.sys*, where *NNN* is a number from 0 to 127.

When specifying a file other than the default, its file specification must contain the string 00, which is incremented by one as subsequent swap files are opened. That allows multiple diskless workstations to swap simultaneously, each using a different swap file.

Periodically, alternate swap files should be deleted from the disk. Those that are in use when you perform the delete operation will not be deleted.

### ***:SwapFileSize:***

CTOS II, CTOS III, and protected-mode CTOS/XE

Default: Current size (workstation); 1500 (SRP)

Minimum: 512 (workstation); 1 (SRP)

Maximum: All available disk space

This parameter specifies the starting size of a swap file. Specify a number of sectors. On virtual memory operating systems, if 0 is specified as the swap file size, a swap file is neither created (if it does not already exist) nor used by the operating system.

### ***:SwapFileSizeMax:***

CTOS II, CTOS III, and protected-mode CTOS/XE

Default: 0 (see below)

Minimum: 0

Maximum: 65535 (workstation)

This parameter controls the maximum size of a disk swap file. Specify a number of sectors, for example, 5000. The swap file size starts at the number of sectors specified for the *:SwapFileSize:* parameter and grows to the size specified for *:SwapFileSizeMax:*. By limiting the size of the swap file, disk space is retained for other uses. Swap files created in memory are not expanded. The default of 0 allows all available disk space to be used.

***:SwapRemoteRunFilesLocally:***

CTOS III

Default: No

This parameter controls whether pages from run files located on the server are copied to the local swap file. Specify **Yes** if you want the paging service to copy pages read from the server to the local swap file and subsequently read them from there. This may improve swapping performance but also may increase the size of the swap file. The default is to read pages from the run file located on the server.

***:UsedFromVirtualRealMode:***

See *:ModuleType:*.

***:VDMFile:***

CTOS II, CTOS III, and CTOS/XE

Default: *[Sys]<Sys>InstallVDM.run*

This parameter determines which video service is installed on a workstation. The default file installs the appropriate video manager for the workstation hardware, as listed below:

Character	<i>[Sys]&lt;Sys&gt;VDM_Ch.run</i>
Bit-mapped	<i>[Sys]&lt;Sys&gt;VDM_Bm.run</i>
VGA (X-Bus and X-Bus+)	<i>[Sys]&lt;Sys&gt;VDM_Vga.run</i>
VGA (Standard EISA/ISA)	<i>[Sys]&lt;Sys&gt;VDM_StdVga.run</i>
VGA (CTOS VideoCard)	<i>[Sys]&lt;Sys&gt;VDM_VgaWei.run</i>

For workstation servers without video, specify the following value for the *:VDMFile:* parameter:

*[Sys]<Sys>VDM\_dmy.run*

On SRPs, this parameter defines the file that installs video services, and it must be located on a volume connected to the master processor.

### **:WakeUpInterval:**

CTOS II and protected-mode CTOS/XE

Default: See below

Minimum: 0

Maximum: 6553

This parameter controls how often the operating system searches for contexts to swap back into memory. Specify a time interval in tenths of seconds. The default value, 0, indicates that swapping occurs only through user action, for example, with the Context Manager.

### **:WatchDogStatus:**

CTOS III and CTOS/XE; master processor only

Default: SetFlag

This parameter defines the action the master processor takes when another processor crashes. If you specify **SetFlag**, unaffected processors continue to run, and on an SRP, the front panel is set to 40.

If you specify **None**, other processors continue to run, and on an SRP, the front panel is not reset. This can cause delays in detecting the crash, and unpredictable errors can occur because interprocessor communications cannot take place with the processor that has crashed. You might, however, want to omit the watchdog when running diagnostics or using the Debugger.

If you specify **Crash**, the master processor forces the entire system to crash. This immediately alerts all users that there is a problem. On an SRP, the front panel status is set to 40 when the watchdog shuts down the system.

**:WBlocks:** (Number = *number*, Size = *bytes*)

CTOS/XE

Default: See subparameters, below

This parameter specifies the number of W-blocks to be allocated on the processor. W-blocks are very large memory buffers used for interprocessor data transfers. Subparameters are described below.

Subparameter	Value
Number	Default: 0 Minimum: 0 Maximum: 1000  Specify the total number of W-blocks to be allocated on the processor.
Size	Specify the number of bytes to be allocated for each W-block.

**:WBlocksIn:** (Number = *number*, Size = *bytes*)

**:WBlocksOut:** (Number = *number*, Size = *bytes*)

CTOS III

Default: See subparameters, below

These parameters define the number and size of W-blocks to be allocated for use by an IOP card. Incoming W-blocks are allocated on the IOP to serve remote requests from the master processor. Outgoing W-blocks are allocated on the master processors to serve requests from the IOP. Add out blocks in small increments and check performance to ensure that you do not inadvertently over subscribe memory.

Subparameter	Value
Number	Default: 2 Minimum: 0 Maximum: 1000  Specify the total number of W-blocks to be allocated on the processor.
Size	Default: 65500 Minimum: Size of Y-block plus 4 bytes Maximum: 65500  Specify the number of bytes to be allocated for each W-block.



**:XBlocks:** (Number = *number*, Size = *bytes*)

CTOS II, CTOS III, and CTOS/XE; processors with cluster lines only; servers only

Default: See subparameters, below

This parameter defines the number and size of large X-blocks to be allocated on the processor. See “Adjusting Memory Blocks,” in Section 15, “Optimizing System Performance,” for information about how X-blocks are used by the operating system. Subparameters are described below.

### Caution

Do not change the value of the *Size* subparameter, described below, unless instructed to do so by a Technical Support engineer. An incorrect value can cause total system failure.

Subparameter	Value
Number	Default: See below Minimum: 1 Maximum: 1000  Specify the number of large X-blocks to be allocated on the processor. The default value is one large X-block per client workstation supported by the operating system. You can use the <b>Cluster Status</b> command to determine the number of currently allocated X-blocks.
Size	Default: 4192 bytes Minimum: 2624 bytes Maximum: 65500 bytes  Specify the number of bytes to be allocated for each large X-block. In addition to the number of bytes specified, a 32-byte (CTOS II and CTOS/XE) or 48-byte (CTOS III) header is allocated for each block.



**:XBlocksSmall:** (Number = *number*, Size = *bytes*)

CTOS II, CTOS III, and CTOS/XE; processors with cluster lines only; servers only

Default: See subparameters, below

This parameter defines the number and size of small X-blocks to be allocated on the processor. Subparameters are described below.

**Caution**

Do not change the value of the *Size* subparameter, described below, unless instructed to do so by a Technical Support engineer. An incorrect value can cause total system failure.

<b>Subparameter</b>	<b>Value</b>
Number	Default: See below Minimum: 0 Maximum: 1000  Specify the number of small X-blocks to be allocated on the processor. The default value is one small X-block per client workstation supported by the operating system. You can use the <b>Cluster Status</b> command to determine the number of currently allocated X-blocks.
Size	Default: 64 bytes Minimum: 0 Maximum: 65500 bytes  Specify the number of bytes to be allocated for each small X-block. (The size specified for small X-blocks should be smaller than the size specified for large X-blocks.) In addition to the number of bytes specified, a 32-byte (CTOS II and CTOS/XE) or 48-byte (CTOS III) header is allocated for each block.

**:YBlocks:** (Number = *number*, Size = *bytes*)

CTOS/XE

Default: See subparameters, below

This parameter defines the number of Y-blocks to be allocated on the processor. Y-blocks are large memory buffers used for interprocessor data transfers. Subparameters are described below.

### Caution

Do not change the value of the *Size* subparameter, described below, unless instructed to do so by a Technical Support engineer. An incorrect value can cause total system failure.

Subparameter	Value
Number	Default: 4 Minimum: 1 Maximum: 64K bytes divided by Y-block size  Specify the total number of Y-blocks to be allocated on the processor.
Size	Default: 4160 bytes Minimum: 2624 bytes Maximum: 64K bytes  Specify the number of bytes to be allocated for each Y-block. In addition to the number of bytes specified, a 32-byte header is allocated for each block.

**:YBlocksIn:** (Number = *number*, Size = *bytes*)

**:YBlocksOut:** (Number = *number*, Size = *bytes*)

### CTOS III

Default: See subparameters, below

These parameters define the number and size of Y-blocks to be allocated for use by an IOP card. Incoming Y-blocks are allocated on the IOP to serve remote requests from the master processor. Outgoing Y-blocks are allocated on the master processors to serve requests from the IOP. Add out blocks in small increments and check performance to ensure that you do not inadvertently over subscribe memory.

Subparameter	Value
Number	Default: 25 Minimum: Value for <i>:nRkvsUsers:</i> plus 1 Maximum: 4000  Specify the total number of Y-blocks to be allocated on the processor.
Size	Default: 4160 Minimum: 2624 bytes Maximum: 65500  Specify the number of bytes to be allocated for each Y-block. In addition to the number of bytes specified, a 32-byte header is allocated for each block.

**:ZBlocks:** (Number = *number*, Size = *bytes*)

### CTOS/XE

Default: See subparameters, below

This parameter defines the number of Z-blocks to be allocated on the processor. Z-blocks are small memory buffers used for interprocessor data transfers. Subparameters are described below.

Subparameter	Value
Number	Default: 28 Minimum: 0 Maximum: 64K bytes divided by Z-block size  Specify the total number of Z-blocks to be allocated on the processor.

Size

Default: 180 bytes  
Minimum: 0  
Maximum: 64K bytes

Specify the number of bytes to be allocated for each Z-block.

**:ZBlocksIn:** (Number = *number*, Size = *bytes*)

**:ZBlocksOut:** (Number = *number*, Size = *bytes*)

### CTOS III

Default: See subparameters, below

These parameters define the number and size of Z-blocks to be allocated for use by an IOP card. Incoming Z-blocks are allocated on the IOP to serve remote requests from the master processor. Outgoing Z-blocks are allocated on the master processors to serve requests from the IOP. Add out blocks in small increments and check performance to ensure that you do not inadvertently over subscribe memory.

#### Subparameter

#### Value

Number

Default: 28  
Minimum: 0  
Maximum: 4000

Specify the total number of Z-blocks to be allocated on the processor.

Size

Default: 180 bytes  
Minimum: 0  
Maximum: 65500

Specify the number of bytes to be allocated for each Z-block.

## Section 17

# Building a Customized Operating System

## When to Customize an Operating System

It is possible for you to customize parameters that are not included in the operating system configuration file. You do not need to be a programmer to build a customized operating system; however, as a system administrator, you are most likely to be carrying out instructions that you receive from someone else, such as a Technical Support engineer. If your questions are not answered in this manual, check with a Technical Support engineer before you create and use a customized operating system.

There are two methods for building a customized operating system, as described below:

### **Mini-Customizer Method**

If you are using a CTOS III operating system, you can customize the number of partitions, process control blocks, user exchanges, messages, file allocation blocks, or file control blocks with the Mini-Customizer.

### **System Build Method**

If you are using CTOS I, CTOS II, or CTOS/XE, or if you need to customize parameters other than those listed above for CTOS III, you need to build (SysGen) a customized operating system by modifying a source file and then assembling and linking the operating system.

Both customization methods are described in detail in this section.

# Installing the System Build Utilities

The Mini-Customizer and the commands and files you need to build an operating system are packaged with the System Build Utilities. Therefore, you need to install the System Build Utilities before you can customize an operating system.

The workstation System Build Utilities are packaged separately from the operating system distribution media. See the CTOS Software Release Announcement for installation instructions.

The SRP System Build Utilities are supplied with the operating system distribution media. See the CTOS/XE Software Release Announcement for installation instructions.

# Using the Mini-Customizer

You can use the Mini-Customizer to permanently increase or decrease the following parameters in CTOS III operating systems:

- Number of partitions
- Number of process control blocks
- Number of user exchanges
- Number of messages
- Number of file allocation blocks
- Number of file control blocks

To use the Mini-Customizer, follow these steps:

1. On the Executive command line, type **Mini Customizer**; then press **GO**.
2. Enter the file specification of the *.img* file you want to customize, as shown in the following example:

```
Enter name of .img file to update
Press GO when ready
vClstrLfs.img
```

---



3. Press **GO**. A form containing the customizable parameters is displayed.
4. Move the cursor to the field you want to customize. (Press **HELP** to display a description of each field.)
5. Press **CODE+DELETE** to clear the field.
6. Type the new value. (If you leave a field blank, its current value is used.)
7. Repeat steps 4, 5, and 6 for each parameter you want to change.
8. Press **GO** to exit the utility and save the file.

## Building an Operating System

Building an operating system includes the following major steps, which are described in detail in this section:

1. Install the the CTOS Development Utilities.
2. Make changes to the source code.
3. Assemble and link the new operating system.

### Installing the Development Utilities

In addition to the System Build Utilities software, you need to install the CTOS Development Utilities. See the Software Release Announcement for that product for installation instructions.

### Making Changes to the Source Code

To make changes to the source code, you use the Editor application. For information about using it, see Section 4, "Using Administrative Tools."

After you install the appropriate software, use the Editor to open and read one of the following files, which contains instructions for customizing operating system parameters:

#### **CTOS I**

*<3.4rGen>rmSysGen.asm*

### CTOS II

*<3.4pGen>SysGen.asm*

### CTOS III

*<1.1vGen>SysGen.asm*

### CTOS/XE

*<Gen>SysGen.asm*

Customizable source code is contained in *prefix files*, which you modify to customize the operating system. Prefix files are named as follows:

*Prefix.asm*

where *Prefix* is a prefix, as listed in Table 17-1 for CTOS I, Table 17-2 for CTOS II, Table 17-3 for CTOS III, and Table 17-4 for CTOS/XE.

For CTOS I, all customizable parameters are contained in a single prefix file. Therefore, regardless of the parameters you need to change, you edit only that single file.

For CTOS II, CTOS III, and CTOS/XE, the file system parameters are in a separate prefix file. In addition, for CTOS II and CTOS III, cluster agent parameters are in yet another prefix file. Therefore, depending on the changes you need to make, you may need to edit more than one file.

To modify a prefix file, you use the Editor application, as described in the following procedure.

1. Use the **Path** command, as shown below, to set your path to the build directory containing the files you want to modify.

Path	
[Volume]	<u>d0</u>
[Directory]	<u>BuildDir</u>
[Default file prefix]	<u></u>
[Password]	<u></u>
[Node]	<u></u>

**Note:** *In the command form shown above, substitute the name of the actual build directory in place of BuildDir. For CTOS I, the build directory is named 3.4rGen; for CTOS II, 3.4pGen; for CTOS III, 1.1vGen; and for CTOS/XE, Gen.*

2. On the Executive command line, type **Editor**; then press **RETURN**.
3. Type the file name of the prefix file you want to modify, as shown in the following example:

```

Editor
File name(s)           pSrvrM.asm
[Read only?]          _____
[Alternate user name]  _____
    
```

4. Press **GO** to open the file.
5. Edit the parameters you want to customize.
6. After you have finished making changes, press **FINISH**, then **GO**, to exit the Editor and save the file.

**Table 17-1. CTOS I Prefix Files**

Prefix	Description
<i>t1Svr</i>	B26 real-mode workstation server
<i>t1Clstr</i>	B26 and CWS real-mode client workstations (diskless)
<i>t1ClstrLfs</i>	B26 real-mode client workstation (local file system)
<i>t1Stnd</i>	B26 real-mode standalone workstation
<i>v1Clstr</i>	B24 real-mode client workstation (diskless)
<i>bawsClstrLfs</i>	B27 real-mode client workstation (local file system)
<i>bawsClstr</i>	B27 real-mode client workstation (diskless)
<i>bawsStnd</i>	B27 real-mode standalone workstation

**Table 17-2. CTOS II Prefix Files**

Prefix	Description
<i>FsS_N</i>	File system for server supporting less than 16 workstations or for client workstation with local file system
<i>FsM_N</i>	File system for server supporting 16 to 24 client workstations
<i>FsL_N</i>	File system for server supporting 24 to 32 client workstations
<i>Srvr</i>	Cluster agent for server
<i>Clstr</i>	Cluster agent for client workstation
<i>pSrvrS</i>	Protected-mode server supporting less than 16 client workstations
<i>pSrvrM</i>	Protected-mode server supporting 16 to 24 client workstations
<i>pSrvrL</i>	Protected-mode server supporting 24 to 32 client workstations
<i>pClstrLfs</i>	Protected-mode client workstation (local file system)
<i>pClstr</i>	Protected-mode client workstation (diskless)
<i>pStnd</i>	Protected-mode standalone workstation

**Table 17-3. CTOS III Prefix Files**

Prefix	Description
<i>FsS_N</i>	File system for server supporting less than 16 workstations or for client workstation with local file system
<i>FsM_N</i>	File system for server supporting 16 to 24 client workstations
<i>FsL_N</i>	File system for server supporting 24 to 32 client workstations
<i>Fs_Srp</i>	File system for multiprocessor servers supporting the maximum configuration of client workstations
<i>Srvr</i>	Cluster agent for server
<i>Clstr</i>	Cluster agent for client workstation
<i>vSrvr</i>	Virtual memory server
<i>vClstrLfs</i>	Virtual memory client workstation (local file system)
<i>vClstr</i>	Virtual memory client workstation (diskless)
<i>vStnd</i>	Virtual memory standalone workstation

**Table 17-4. CTOS/XE Prefix Files**

Prefix	Description
<i>Fs_Si</i>	File system for General Processors with SCSI Interface
<i>Fs_Fp</i>	File system for File Processors
<i>Fs_Dp</i>	File system for Data Processors
<i>pSrpGp</i>	Protected mode General Processors
<i>rSrpFp</i>	Real mode File Processors
<i>rSrpCp</i>	Real mode Cluster Processors
<i>rSrpTp</i>	Real mode Terminal Processors
<i>rSrpDp</i>	Real mode Data Processors
<i>rSrpSp</i>	Real mode Storage Processors

## Assembling and Linking

After you have made changes to the prefix file(s), you assemble and link them to build a new operating system. Procedures vary according to the prefix files you customized, as described in the following sections.

### File System Prefix File

*Note:* If you are customizing a CTOS I operating system, or if you did not make changes to the file system, skip this section.

To assemble and link the file system, follow these steps:

1. On the Executive command line, type **Assemble**; then press **RETURN**.
2. Enter the name of the file system prefix file, as shown in the following example:

```
Assemble
File name      Fsm N.asm
[Errors only?] yes
```

3. Press **GO**.

If no error messages occur during the assembly operation, proceed to the next step. If errors occur, see "Troubleshooting SysGen Errors," later in this section.

4. On the Executive command line, type **Link File System**; then press **RETURN**.
5. Fill in the command form, as shown in the following example.

```
Link File System
File system type (e.g. Fsm_N)  Fsm N
Version                        3.4-7/30
[File system name (FileSys)]
```

Notice that you can include a date or other brief description to differentiate the customized file system from the prebuilt version.

6. Press **GO**.
7. Assemble and link the cluster agent and or the operating system prefix file, as described in the sections that follow.

### Cluster Agent Prefix File

*Note: If you are customizing a CTOS I or CTOS/XE operating system, or if you did not make changes to the cluster agent, skip this section.*

To assemble and link the cluster agent, follow these steps:

1. On the Executive command line, type **Assemble**; then press **RETURN**.
2. Enter the name of the cluster agent prefix file, as shown in the following example:

```
Assemble
File name          Srvr.asm
[Errors only?]    yes
```



3. Press **GO**.

If no error messages occur during the assembly operation, proceed to the next step. If errors occur, see "Troubleshooting SysGen Errors," later in this section.

4. On the Executive command line, type **Link Agent**; then press **RETURN**.
5. Fill in the command form, as shown in the following example.

```
Link Agent
  Agent type (Srvr or Clstr)  Srvr
  Version                    3.4-7/30
```

Notice that you can include a date or other brief description to differentiate the customized agent from the prebuilt version.

6. Press **GO**.
7. Assemble and link the operating system prefix file, as described in the next section.

### Operating System Prefix File

To assemble and link the operating system prefix file, follow these steps:

1. On the Executive command line, type **Assemble**; then press **RETURN**.
2. Enter the name of the prefix file for the operating system you want to build, as shown in the following example.

```
Assemble
  File name          pSrvrM.asm
  [Errors only?]    yes
```

Notice that you can include a date or other brief description to differentiate the customized operating system from the prebuilt version.

3. Press **GO**.

If no error messages occur during the assembly operation, proceed to the next step. If errors occur, see "Troubleshooting SysGen Errors," later in this section.

4. On the Executive command line, type the appropriate command name from the list below; then press **RETURN**.

**Link CTOS I** (real-mode workstation)

**Link CTOS II** (protected-mode workstation)

**Link CTOS III** (virtual-memory workstation)

**Link CTOS** (real-mode SRP)

**Link CTOS VM** (protected-mode SRP)

5. Enter the operating system prefix and a version number, as shown in the following example:

```
Link CTOS II
  Operating system type (e.g. pClstr)  pSrvrM
  Version                               3.4-7/30
```

6. Press **GO**.

## Testing the New Operating System

On a workstation, you use the **Bootstrap** command to test the new operating system before you copy it to `[Sys]<Sys>SysImage.sys`. On an SRP, you use different methods for the master processor and the other processors. All methods are described below.

When the new operating system is running on the workstation or SRP, try to reproduce the problems or inadequacies you were experiencing with the original operating system. Keep track of whether or not they improve.

If the operating system will not boot, see “Troubleshooting SysGen Errors,” later in this section.

### On a Workstation

To load the new operating system, use the **Bootstrap** command, as described in the following procedure:

1. Make sure your path is still set to the appropriate build directory.
2. On the Executive command line, type **Bootstrap**; then press **RETURN**.

3. Specify the name of the operating system file. After build, operating systems are named with the operating system prefix and one of the following suffixes:
  - *.img* for protected-mode operating systems
  - *.run* for real mode-operating systems

For example:

```
Bootstrap
File name           [d0]<3.4pGen>pSrvrM.img
[Sys volume or wsNNN]
[OS configuration file]
```

4. Press **GO**.

### On an SRP Master Processor

To test an operating system on the master processor of an SRP, you can copy it to a QIC tape and then bootstrap from the tape.

To copy the customized operating system to a QIC tape, follow these steps:

1. Insert a blank QIC tape into the tape drive on the SRP.
2. On the Executive command line, type **Tape Copy**; then press **RETURN**.
3. Fill in the command form, as shown in the following example:

```
Tape Copy
File from           [d2]<Gen>pSrpGp.img
File to             [QIC] 0
[Overwrite ok?]
```

4. Press **GO** to copy the file.

To bootstrap from the tape, reset the SRP to the M (manual) keyswitch position.

If you prefer, you can boot an SRP master processor with the **Bootstrap** command via Cluster View. In some cases, though, real mode SRP processors do not contain enough memory to execute the **Bootstrap** command.

After the SRP boots, you can use the Partition Status command via Cluster View to make sure that it has booted the customized operating system.

### On Other SRP Processors

To test the new operating system on SRP processors other than the master processor, edit the operating system configuration file, as shown in the following example. Note that the value of the OS subparameter is the file specification of the customized operating system.

```
:Boot: (Processor = GP01, OS = [d2]<Gen>pSrpGp.img)
```

See Section 16, “Configuring Operating Systems,” for more detailed information about the operating system configuration file.

## Installing the New Operating System

On workstations and SRP master processors, the new operating system bootstraps on a one-time basis with the methods described above. If you turn off or reset the system, the original operating system bootstraps from *[Sys]<Sys>SysImage.sys*. Once you are satisfied that the new operating system is functioning correctly, you can permanently install it in *[Sys]<Sys>SysImage.sys*.

To install the new operating system into *[Sys]<Sys>SysImage.sys*, use the **Copy** command, as shown in the following example:

```
Copy
File from      [d0]<3.4pGen>pSrvrM.img
File to        [Sys]<Sys>SysImage.sys
[Overwrite ok?] yes
[Confirm each?]
```

## Troubleshooting SysGen Errors

If you have made inappropriate changes or typographical errors when editing a prefix file, errors will occur during the assembly or link operations.

### Assembly Errors

The most common cause of errors during the assembly operation is that you have specified a value that is too large for the parameter field, or that you have accidentally deleted the punctuation marks that enclose comments.

With the **Editor** or the **Type** command, examine the *.lst* file that is created during the assembly operation. This file contains a list of errors that occurred while the source file was being assembled. Such files are named as follows:

*Prefix.lst*

where *Prefix* matches the prefix of the source file you are attempting to assemble. (See Table 17-1 for CTOS I, Table 17-2 for CTOS II, Table 17-3 for CTOS III, or Table 17-4 for CTOS/XE.)

Reedit the source file to fix the error; then reassemble the source file with the **Assemble** command. Repeat this process until the assembly operation is executed with no errors.

### Link Errors

The most common cause of errors during the link operation is not having all the required files in your build directory. This can happen if you did not use the proper command to install the System Build Utilities software, or if some files have been accidentally deleted from the build directory.



If this should happen, examine the *.map* file that is created during the link operations. This file contains information about which files are missing or invalid during the link operation. Such files are named as follows:

*Prefix.map*

where *Prefix* matches the prefix of the file you are attempting to link. (See Table 17-1 for CTOS I, Table 17-2 for CTOS II, Table 17-3 for CTOS III, or Table 17-4 for CTOS/XE.)

To correct link errors, copy the required files from the distribution media into your build directory; then execute the link operation again.

### Bootstrap Errors

Some errors that prevent the operating system from booting are not detected during assembly and linking. Bootstrap errors usually occur because you have allocated too much memory for certain parameters, particularly file system parameter values, such as *nFab* (file allocation blocks) and *nVhb* (volume home blocks).

If this should happen, make the parameter values smaller by editing the prefix file again; then rebuild the operating system.



# Section 18

## Customizing Standard Software

### What You Can Customize

For Standard Software and many other applications, you can customize messages and prompts that appear on the screen. The following components of Standard Software can be customized:

- Message files, which contain prompts and error messages for most Standard Software commands
- Template files, which contain menus and function key labels for certain commands only
- The SignOn messages for both the conventional and window-interface screens

Customization is most frequently performed to translate messages into other languages. However, it can also be performed to clarify or change the wording of English messages. Be aware, though, that significant changes to screen messages can make it difficult for users to follow the published documentation.

### Customizing Message Files

Screen messages are contained in *binary message files*. Binary message files contain machine-readable code, which is interpreted by programs to display messages. Messages for Standard Software are contained in a number of different message files. See the Software Release Announcement for a list of message files and the commands to which they pertain.

Major customization of message files requires a Product Localization Kit (PLK) for the product you want to customize. It is possible, though, to make minor changes to the Standard Software message files, such as adding a customized heading to the SignOn screen (described later in this section).

This section contains procedures for customizing screen messages. See also the *CTOS Executive Reference Manual* for more information about the **List Message File** and **Create Message File** commands.

### Generating a Message Text File

The binary message file does not contain readable text. You can, however, generate an ASCII output file that translates a binary message file into readable text. Such a file is called a *message text file*.

To generate a message text file, follow these steps:

1. On the Executive command line, type **List Message File**; then press **RETURN**.
2. Fill in the command form, as shown in the following example:

```
List Message File
  Binary file  [Sys]<Sys>ExecMsg.bin
  [Text file]  _____
```

In the *Binary file* field, enter the name of the message file you want to customize.

You can leave the *[Text file]* field blank. By default, the text file name is the same as the binary file, except for its suffix. Listed text files end with *.txt*, while binary files end with *.bin*.

3. Press **GO**.

### Editing a Message Text File

To make changes to a message text file, open the message text file with the Editor, as shown in the following example:

```
Editor
  [File name(s)]      [Sys]<Sys>ExecMsg.txt
  [Read only?]
  [Alternate user name] _____
```

See Section 4, "Using Administrative Tools," for information about using the Editor.

Observe the following rules and guidelines as you edit a message text file:

- Change only messages and prompts enclosed in quotation marks ("").
- Do not change numbers surrounded by colons (:).
- Do not change characters preceded by a percent sign (%).
- Do not change file specifications.
- Keep new messages approximately the same length as the original messages.
- Add comments by preceding text with a semicolon (;).

### Creating a Binary Message File

After you have edited the message text file, you create a new binary message file. To do so, use the **Create Message File** command, as described in the following procedure.

#### Caution

Before you begin the following procedure, make a copy of the original binary message file. By doing so, you can easily restore the original version, if necessary. For example, you could copy *ExecMsg.bin* to a file named *CustomExecMsg.bin*, or a similar descriptive name. Do not, however, rename any of the message files. The exact file names, as listed in the Software Release Announcement, are required for message files to work properly .

1. On the Executive command line, type **Create Message File**; then press **RETURN**.
2. Fill in the command form, as shown in the following example:

Create Message File

Text file [Sys]<Sys>ExecMsg.txt

[Message file] \_\_\_\_\_

[Print file] \_\_\_\_\_

In the *Text file* field, specify the name of the message text file that you modified with the Editor.

You can leave the *[Message file]* field blank. By default, the binary file name is the same as the text file, except for its suffix. Binary message files end with *.bin*, while text files end with *.txt*.

### 3. Press GO.

After the command has been executed, changes in the new binary message file are automatically implemented. You do not need to log out or reboot the workstation.

## Merging Message Files

When you update software on your system, you can merge customized message files with those that are newly released. This preserves the customized files you have created but merges new messages into them. See the **Merge Binary Message Files** and **Merge Message Files** commands, in the *CTOS Executive Reference Manual*, for more detailed information.

## Customizing Templates

The **System Manager** and **User File Editor** commands use template files to display menus and function-key labels. What you see on the screen can be changed by modifying the template file.

Template files for the **System Manager** and **User File Editor** commands are named as follows:

*[Sys]<Sys>SystemMgrConfig.sys*

*[Sys]<Sys>UserFileTemplate.sys*

Template files are written in the *:Keyword:Value* format, which is described for other configuration files throughout this manual. (For an example, see Section 7, "Customizing User Environments.") To make changes to a template file, you use the Editor application.

For detailed information about template file formats and keywords, see the **System Manager** and **User File Editor** commands, in the *CTOS Executive Reference Manual*.

## Customizing the SignOn Screen

By default, the conventional SignOn screen, as shown in Figure 18-1, is displayed after a workstation bootstraps. Optionally, you can configure a workstation to display the window-interface SignOn screen, as shown in Figure 18-2. In addition, you can customize the SignOn screen to display messages that are appropriate for your workplace. Procedures for customizing the SignOn display are included in this section.

**Figure 18-1. SignOn Screen: Conventional Interface**

SignOn 12.3.0 (OS CTOS III 1.1.0)

This message can be customized.

Tue Jul 27, 1993 10:25 AM

User name	Enter a user name, an application name or leave this line blank to display a command prompt.
Password	Enter your assigned password (optional).
Day/Date/Time	Enter the current day, date and time (if not already set).
Environment	Enter the name of the user environment (optional).

Then press the GO key.

User name (e.g. Pam)

Password

Date/Time (e.g. Fri Dec 13, 1991 10:00am)

User environment

603.18-1



Figure 18-2. SignOn Screen: Window Interface

SignOn

User name	Enter a user name, an application name or leave this line blank to display a command prompt.
Password	Enter your assigned password (optional).
Day/Date/Time	Enter the current day, date and time (if not already set).
Environment	Enter the name of the user environment (optional).

Select LOGIN to display the SignOn form.

603.18-2

### Activating the Window-Interface SignOn Screen

To activate the window-interface SignOn program, the XVT system service must be installed on the workstation. XVT is not included with Standard Software but is packaged and installed separately. See the Software Release Announcement for XVT for installation instructions. For information about installing software, see Section 8, "Installing Applications," in this manual.

**Note:** *Some workstations do not contain enough memory to display the window-interface SignOn screen, even though the XVT software has been properly installed. In such cases, the conventional SignOn screen is displayed, and it is not unusual to notice a delay before it appears.*



## Disabling the Window-Interface SignOn Screen

In some cases, you may need to install the XVT system service for other applications that require it but prefer to use the conventional SignOn screen. To disable the window interface SignOn screen, delete or rename the file named *[Sys]<Sys>XVTSignOn.run*. If that file is not present, the conventional SignOn screen is displayed.

## Editing the SignOn Heading

On the conventional SignOn screen, a heading is displayed above the double bars (see Figure 18-1). It contains the version number of the SignOn program and the system date and time. Below the version number, 50 characters of customized text can be displayed on a single line. For example, this line could be used to display the name of a company or a network node.

To customize the SignOn heading, perform the steps listed below. See “Customizing Message Files,” earlier in this section, for detailed procedures for creating and editing message files.

1. Create a message text file from *[Sys]<Sys>SignOnMsg.bin*.
2. Edit line 505 of that message text file to contain the 50-character line of text you want to display.
3. Create a new version of *[Sys]<Sys>SignOnMsg.bin*.

The customized message is displayed on the conventional SignOn screen the next time you log out.

## Editing the SignOn Text File

The text displayed on the SignOn screen is contained in an editable text file, named *[Sys]<Sys>SignOn.txt* for the conventional screen or *[Sys]<Sys>XVTSignOn.txt* for the window-interface screen. You can edit either SignOn text file to display a message that is appropriate for your workplace.

In addition, you can create customized SignOn text files for client workstations that boot from server. Such files must be located in *[Sys]<Sys>* on the server and are checked in the following order:

### Conventional SignOn

*[Sys]<Sys>HwNNN>SignOn.txt*

*[Sys]<Sys>WsNNN>SignOn.txt*

*[Sys]<Sys>Ws>SignOn.txt*

### Window-Interface SignOn

*[Sys]<Sys>HwNNN>XVTSignOn.txt*

*[Sys]<Sys>WsNNN>XVTSignOn.txt*

*[Sys]<Sys>Ws>XVTSignOn.txt*

(*NNN* is the three-digit hardware ID or workstation-type number.)

If none of the files listed above exists, *SignOn.txt* or *XVTSignOn.txt* is displayed.

See Section 5, "Bootstrapping," for more information about hardware IDs and workstation-type numbers.

## Setting Color Attributes

The Executive and other applications read the system color palette, which in most cases can be controlled with a command from within the application. In the Executive, you use the **Screen Setup** and **Video** commands to set the color palette. Whether those colors are retained when you start another application depends on how color has been implemented programmatically by that application.

See the **Screen Setup** command, in the *CTOS Executive Reference Manual*, for information about setting the color palette for the Executive. See the *CTOS Programming Guide* for information about programming the system color palette.

## Displaying Color on the SignOn Screen

When you log out, the SignOn program resets character color attributes to green. To display color on the SignOn screen, you need to include video escape sequences for color attributes within the SignOn text file. See the documentation for the **Submit** command, in the *CTOS Executive Reference Manual*, for information about using video escape sequences.

Each position in the color palette is represented by a hexadecimal number. For SignOn, those hexadecimal numbers correspond to preselected colors, as listed below:

*Note:* Because the color palette can be reprogrammed by other applications, the colors listed below may not apply to applications other than SignOn.

Color	Hexadecimal Number
Green	00
Dark green	01
Blue	40
Violet	41
Red	80
Orange	81
White	C0
Yellow	C1

After you add color attributes to a text file, use the **Type** command to display the contents of the file.

## Mixing Color and Other Video Attributes

To mix color with other attributes, such as blinking and bold, you add the hexadecimal values for all attributes and include that total value in the video escape sequence. See the **Submit** command, in the *CTOS Executive Reference Manual*, for details.



# Section 19

## Troubleshooting

### Diagnosing Problems

As a system administrator, you are responsible for isolating and correcting a wide variety of problems. The following commands, which are described in this section, are useful tools for diagnosing problems:

- **PLog**
- **Cluster Status**
- **Partition Status**

See also the *CTOS Status Codes Reference Manual* and your hardware and software applications manuals for more detailed information about specific problems.

#### **PLog**

You use the **PLog** command to view the system log file. It is a good idea to read the log file regularly, even if no specific problems occur. This can help you to identify marginally functioning hardware, and many applications record messages and errors in the system log file.

The following events and errors are routinely recorded in the system log file:

- System bootstrap events
- System initialization errors
- System crashes
- Disk errors
- Cluster communication errors
- ISAM errors

To start **PLog**, type the command name on the Executive command line; then press **GO**.

With **PLog**, you can optionally view a selected group of errors, based on the error type or date. You can also print the log file. See also the **PLog** command in the *CTOS Executive Reference Manual* for more information.

Each entry in the system log file contains the date and time of occurrence, type, and additional information about the error or event. As the log file fills, newer entries replace the older entries.

The following examples show some system log file entries.

### Example 1: System Bootstrap Event

The following example shows an entry for a system bootstrap event. It records the type of workstation, amount of memory, date and time of the event, and version of the operating system that was booted.

```
NGENT3, Cluster Workstation, With File System
Memory Size: 3584K, SignOn User Name: ...
SYSTEM BOOT - Tue Jul 6, 1993 2:35 PM
Os Booted: pClstrLfs 3.4.0
```

### Example 2: Disk Error

The following example shows an entry for a disk error. Notice that this error was recovered; this means that no damage occurred, but such an error can signal impending problems with a disk.

```
NGENT3, Cluster Workstation, With File System
Memory Size: 3584K, SignOn User Name: Tricia
DISK ERROR - Winchester Unit 0 (ERC = 301)
Description: CRC error in data field
Number of Retries: 2 (Recovered), Volume Name: TF1
Cylinder: 418, Head: 0, Sector: 11, Num of Sectors: 5
Command: 2D
Main Status: 51, Error Status: 40
```



### Example 3: System Initialization Error

The following example shows an entry for a system initialization keyboard error.

```
386i, Server Workstation, No CommIop
Memory Size: 4096K, SignOn User Name: ...
SYSTEM INITIALIZATION ERROR -
Description: No keyboard hardware
Initialization Status: 0080H
```

### Example 4: Processor Crash

The following examples show entries for workstation and SRP processor crashes. Use the **Status Codes Reporter** command to obtain for a description of the error, for example, ERC=89; the executing instruction and crash information may be useful to a Technical Support engineer or system programmer.

#### Workstation

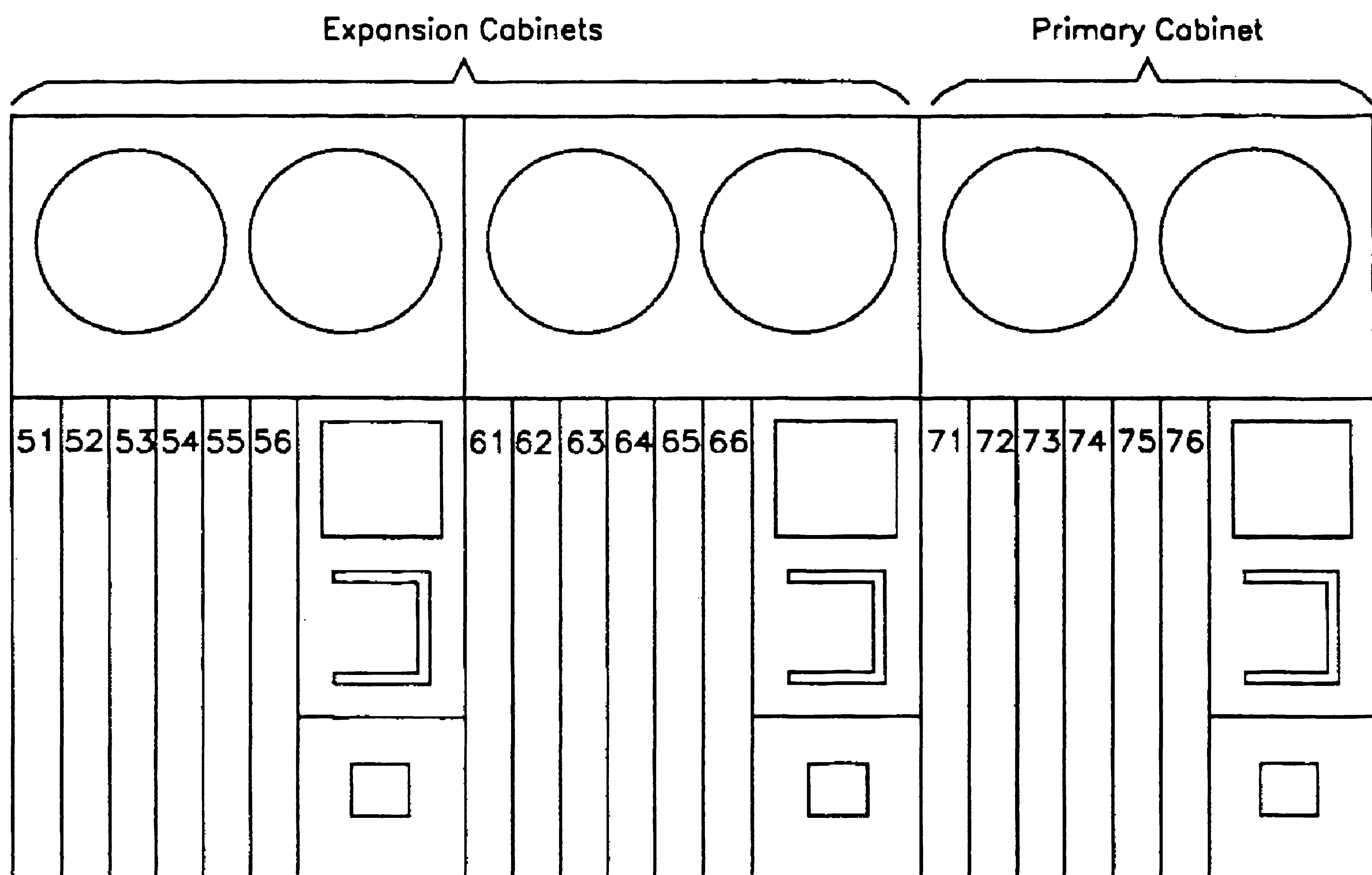
```
NGENT3, Cluster Workstation, With File System
Memory Size: 3584K, SignOn User Name: Tricia
SYSTEM CRASH - (ERC = 89)
Executing instruction preceding location 02F0:04AC
Crash Information:
0059H 0000H 0000H 0000H 8006H 1900H 02F0H 04ACH
Os Booted: pClstrLfs 3.4.0
```

#### SRP

```
Cluster Processor in slot 73H
Memory Size: 768K
SYSTEM CRASH - (ERC = 8105)
Executing instruction preceding location
0C84:04B2
Crash Information:
1FA9H 0005H 0000H 0000H 0073H 0000H 0C84H 04B2H
Os Booted: rSrpCp-3.4.0
```

Note that entries for SRP processors reference a type of processor and a slot number. The slot number identifies the processor where the error occurred. Slots in additional expansion cabinets continue the same numbering scheme. Figure 19-1 shows slot numbers for a three-cabinet SRP.

Figure 19-1. SRP Slot Numbers (in Hexadecimal)



603.19-1

## Cluster Status

The **Cluster Status** command provides information about cluster communications. It displays the SignOn user name of workstations connected to the cluster and lists cluster communication errors. Although the information displayed by **Cluster Status** is rather technical, certain types of errors can indicate particular problems.

To start **Cluster Status**, type the command name on the Executive command line; then press **GO**. On a multiprocessor workstation or SRP, use **F10** (Next), after the command is started, to cycle through the cluster lines.

See "Workstation Troubleshooting," later in this section, for information about correcting cluster communication errors. See also the *CTOS Executive Reference Manual* for more detailed information about the **Cluster Status** command.

## Partition Status

The **Partition Status** command provides information about the random access memory (RAM) on a processor. It is used to find out how memory is being used and how much remains available. System administrators use this command most frequently to obtain the following information:

- Whether system services are installed
- How much memory remains available on a processor
- Whether a portion of memory is not functioning

Instructions for starting **Partition Status** and a description of the screen are contained in “Calculating Memory Requirements,” in Section 9, “Installing System Services.” See also the **Partition Status** command in the *CTOS Executive Reference Manual*.

## Workstation Troubleshooting

This section provides information about common hardware and software problems. It is organized by symptom, as follows:

- Workstation power does not come on.
- Workstation does not bootstrap.
- Module or card is not recognized.
- Keyboard does not work.
- Video does not work properly.
- Workstation does not communicate with the server.
- Application cannot be started.
- Application is running slowly.

### Workstation Power Does Not Come On

Use the following list to identify and correct problems that prevent the workstation from turning on.

---

<b>Possible Cause</b>	<b>Corrective Action</b>
Power supplies might not be plugged in.	Check plugs, power cords, and the jumper cords that connect power supplies.
Power supply might be broken.	On X-Bus workstations with external power supplies (bricks), replace the power supply with one that you know is working.
Wall outlet might not be supplying power.	Test with an outlet that you know supplies power.

---

## Workstation Does Not Bootstrap

Use the following list to identify problems that prevent the workstation from booting. Several of these problems are described in more detail later in this section. (See "What If a System Will Not Bootstrap?," later in this section.)

Possible Cause	Corrective Action
System power might not be turned on.	Turn on the workstation. See also "Workstation Power Does Not Come On," above.
The disk containing the system volume might not be recognized.	See "Module or Card Is Not Recognized," below.
The system volume might be corrupted.	See "What If a System Will Not Bootstrap?," later in this section.
System software might be incorrectly or incompletely installed.	See the <i>CTOS System Software Installation Guide</i> and the Software Release Announcements for your system software products.
A customized operating system might not be bootable.	See Section 18, "Building a Customized Operating System" and "What If a System Will Not Bootstrap?," later in this section.
Hardware might be installed incorrectly.	See the appropriate hardware installation manual.
Processor or memory hardware might be failing.	Run diagnostics on the processor and memory; see the manual for diagnostics software.

## Module or Card Is Not Recognized

Use the following list to identify and correct problems that prevent a particular module or expansion card from being recognized on a workstation that is otherwise functioning correctly.

---

Possible Cause	Corrective Action
Modules might be improperly positioned.	X-Bus and SCSI modules must be properly positioned; otherwise, some of them will not be recognized. In many cases, X-Bus modules must be placed immediately to the right of the processor, followed by SCSI modules. See your hardware installation guides, however, for more detailed information.
Too many modules might be connected to the workstation.	The X-Bus length cannot exceed 24 inches. (X-Bus length is the distance between the first X-Bus connection and the last, as shown in the following illustration. Turn off the workstation and disconnect power. Remove any modules that make the X-Bus too long.
	X-Bus Length
	543.20-A
	In addition, the number of SCSI modules is limited, depending on the workstation model. In many cases, that limitation is one SCSI Upgrade module, followed by six SCSI Expansion modules. See your hardware installation guide, however, for more detailed information.
More power supplies might be needed.	Each power supply can accommodate only a certain number of power units. See the installation guide for your workstation.

---

continued



continued

Possible Cause	Corrective Action
A power supply might be broken.	Replace the power supply with one that you know is working.
Modules or cards might not be properly seated.	<p>On X-Bus workstations, turn off the workstation and disconnect power. Disconnect the offending module, and check for bent or broken pins. If the pins appear to be undamaged, reconnect each module, pressing the X-Bus firmly together before latching.</p> <p>On EISA/ISA workstations, remove the workstation cover and check placement of the card that isn't recognized.</p> <p>If pins are broken or bent, have the module or card repaired before replacing it on the workstation.</p>
On EISA/ISA workstations, cards might not be properly configured for recognition on the system.	Run the Setup program on ISA workstations) or the System Configuration Utility (SCU) on EISA workstations. See the OpenSystems Workstations hardware installation guide for your workstation.
IOP card might be installed in an ISA workstation.	IOPs are EISA cards and although they may power up on an ISA system, they do not work. Install IOP cards in EISA systems only.
On an EISA system, an IOPs boots, but the system hangs with no video.	Interrupt switch selector on the IOP must select 15 (sw1 ON, sw2, 3, and 4 OFF).

### Keyboard Does Not Work

Use the following list to identify and correct problems that prevent the keyboard from working.

---

Possible Cause	Corrective Action
The keyboard cable might be disconnected.	Check the attachment to the keyboard and to the monitor.
The keyboard might be broken.	Replace with a keyboard that you know is working.
The keyboard outlet on the monitor might be broken.	Replace with a monitor that you know is working.
The Remote User Manager might be installed.	Edit the system initialization file to remove the entry that installs the Remote User Manager ( <i>RUM.run</i> ) or that installs the Login Service ( <i>Login.run</i> ); then reboot the workstation. See "Installing the Remote User Manager," in Section 9, "Installing System Services."
An I-bus adapter might be needed.	EISA and ISA workstations require an I-bus adapter to use Unisys keyboard models, such as the K1 and K5.

---

## Video Does Not Work Properly

Use the following list to identify and correct video problems.

Possible Cause	Corrective Action
The On/Off switch might need to be turned on.	Some monitors have an On/Off switch (see the installation guide or owner's manual for the monitor). Turn on the On/Off switch and wait a few moments for the monitor to warm up.
The brightness control might be turned down.	Adjust the brightness control knob to increase brightness. (See the installation guide for the monitor.)
The monitor cable might be loose or disconnected.	Check its connection to the workstation or video controller.
The correct Video Access Method (VAM) software might not be installed on the workstation.	Install the VAM software.
The monitor might be connected to the wrong port.	If the workstation has a graphics module or a graphics card, the monitor must be attached to it rather than to the port on the processor module.
The graphics controller might be incompatible with the monitor.	The graphics controller on the workstation must support the type of monitor that is attached to the workstation. See the installation guide for the workstation processor.
The monitor might be broken.	Replace it with a monitor that you know is working.
On EISA/ISA workstations, the CTOS VideoCard might not be properly configured.	Run Setup or the System Configuration Utility and set the interrupt level to 11.
On an IOP, the correct Video Access Method might not be installed.	Install character video from the VAM installation media.

### Workstation Does Not Communicate With the Server

Use the following list to identify and correct problems that prevent the workstation from communicating with the server. See the *CTOS Cluster and Network Hardware Installation Guide* for more information about cluster communication problems.

---

Possible Cause	Corrective Action
The server might be turned off or might have crashed.	Reboot the server.
Cluster cables might be disconnected.	Check connections both at the workstation and at the server.
A workstation in the cluster might not be properly terminated.	Check both ends of the daisy chain and, if required, install a terminator (see the <i>CTOS Cluster and Network Hardware Installation Guide</i> ).  With TeleCluster, make sure that each workstation is terminated.
On EISA servers, the cluster line might be plugged into the Client Card, rather than the Cluster IOP.	Move the cluster cable to the Cluster IOP.
Too many workstations might be connected to the server.	Use the <b>Cluster Status</b> command to determine the total number of workstations the server supports and how many are already active.  To support more workstations, you may need a different server or a customized operating system.
A protocol error might have occurred on a client workstation.	Reboot the malfunctioning workstation.
A client workstation might be running a server operating system.	This may not be the workstation that is receiving the error. Check the Executive status area on each workstation and install the correct operating system on the offender. You may need to reboot the server.
If using TeleCluster, the hub might be malfunctioning.	Call for service from your telephone service company or department.

---

## Application Cannot Be Started

Use the following list to identify and correct problems that prevent applications from starting. If an error code is displayed, write it down in case you need to call Technical Support.

Possible Cause	Corrective Action
The software installation procedure might not have been performed correctly or completely.	Repeat the installation procedure by using the <b>Installation Manager</b> command. Never perform a partial installation, such as just copying the run files, of applications products.
A required system service might not be installed.	Check the release documentation for information about the system services that are required to run the application.
The disk might be full.	Check disk space with the <b>Volume Status</b> command. Perform a disk cleanup (see the <i>CTOS Executive User's Guide</i> ) or install the application on a different volume. Consider adding or upgrading to a larger disk.
The installation might have failed because a password is required.	Enter the volume password (use the <b>Path</b> command as described in the <i>CTOS Executive User's Guide</i> ) and repeat the installation procedure.
A password might be required to start the application.	Enter a valid password before starting the application.
System software might require updating.	Check the release documentation to determine the software requirements for the product.

### Application Is Running Slowly

Use the following list to identify and correct problems that cause poor performance on your workstation.

---

Possible Cause	Corrective Action
The application might be running in a small partition.	<p>Use the <b>Partition Status</b> command to determine the size of the partition and to determine whether additional memory is available. If using Context Manager, increase the partition size in the Context Manager configuration file. See Section 15, "Optimizing System Performance."</p> <p>If the workstation is not running Context Manager, deinstall unnecessary system services.</p> <p>Consider expanding memory on the workstation.</p>
The system volume might be more than 90% full.	<p>Use the <b>Volume Status</b> command to find out how much disk space has been used (see the <i>CTOS Executive User's Guide</i>). Perform a disk cleanup procedure to remove unnecessary files. Then, back up the disk and use <b>Disk Squash</b> to compress fragmentation or reinitialize the disk and restore the backup.</p> <p>If a printer is attached to the workstation, remove temporary print files from the &lt;GPS&gt; directory (see the <i>CTOS Generic Print System Administration Guide</i>).</p> <p>Consider adding another disk.</p>
A portion of memory might be malfunctioning.	<p>See the hardware diagnostics manual for the workstation.</p>
On a virtual memory system, applications may be contending for oversubscribed memory.	<p>See "Optimizing Virtual Memory," in Section 15, "Optimizing System Performance."</p>
On a multiprocessor workstation, the application might be running on an IOP.	<p>Exit the application, exit the Cluster View session, and restart the application on the master processor.</p>

---



## SRP Troubleshooting

A number of special considerations for troubleshooting pertain to the SRP. This section contains general information about identifying problems and correcting them. See the installation guides and the hardware diagnostics manual for more detailed information.

### Hardware Installation Problems

Many problems associated with the SRP can be traced back to hardware installation. Common hardware installation problems are listed below. See the appropriate installation guide for information about correcting them:

- Seating the boards incorrectly
- Leaving vacant slots between boards within an enclosure
- Installing disk drives improperly
- Placing strapping jumpers incorrectly on memory expansion boards
- Failing to terminate cluster lines (see the *CTOS Cluster and Network Hardware Installation Guide*)

### Processor Crashes

The SRP provides the following tools for troubleshooting a processor crash:

- A two-digit front panel display
- An LED display on the back of each processor

This section describes how to read and interpret the status information provided by these displays.

### Interpreting the Front Panel Display

The front panel displays a two-digit number to indicate system status, as listed below. See the *CTOS Status Codes Reference Manual* for more detailed information.

00 to 05	Occur during the normal bootstrap sequence when the bootstrap ROM is executing.
06 to 10	Indicate that the bootstrap ROM has completed its job and the operating system has taken over the system initialization sequence.
20	Indicates that the system is running and functioning normally.
21 to 29	Indicate that no bootable operating system was located by the bootstrap ROM.
30 to 39	Indicate a hardware error detected by the bootstrap ROM.
40	Indicates that the watchdog on the master processor has detected a processor crash.
50 to 51	Indicate an error during system service installation.

### Interpreting Processor LEDs

When an SRP crashes, the front panel display will usually report the condition. If, however, a watchdog is not set (see the description of that parameter, in Section 16, "Configuring Operating Systems"), the front panel may continue to display 20 if a processor, other than the master processor, crashes. Processor crashes can always be detected by examining the processor status lights.

Crash codes are displayed on a processor after it has crashed but before the system is reset. It is important to read and interpret the crash code *before* you reboot the system. Therefore, if you are troubleshooting a recurring processor crash, bootstrap the system from the *M* (manual) keyswitch position so that the processor will not be rebooted automatically after it crashes.

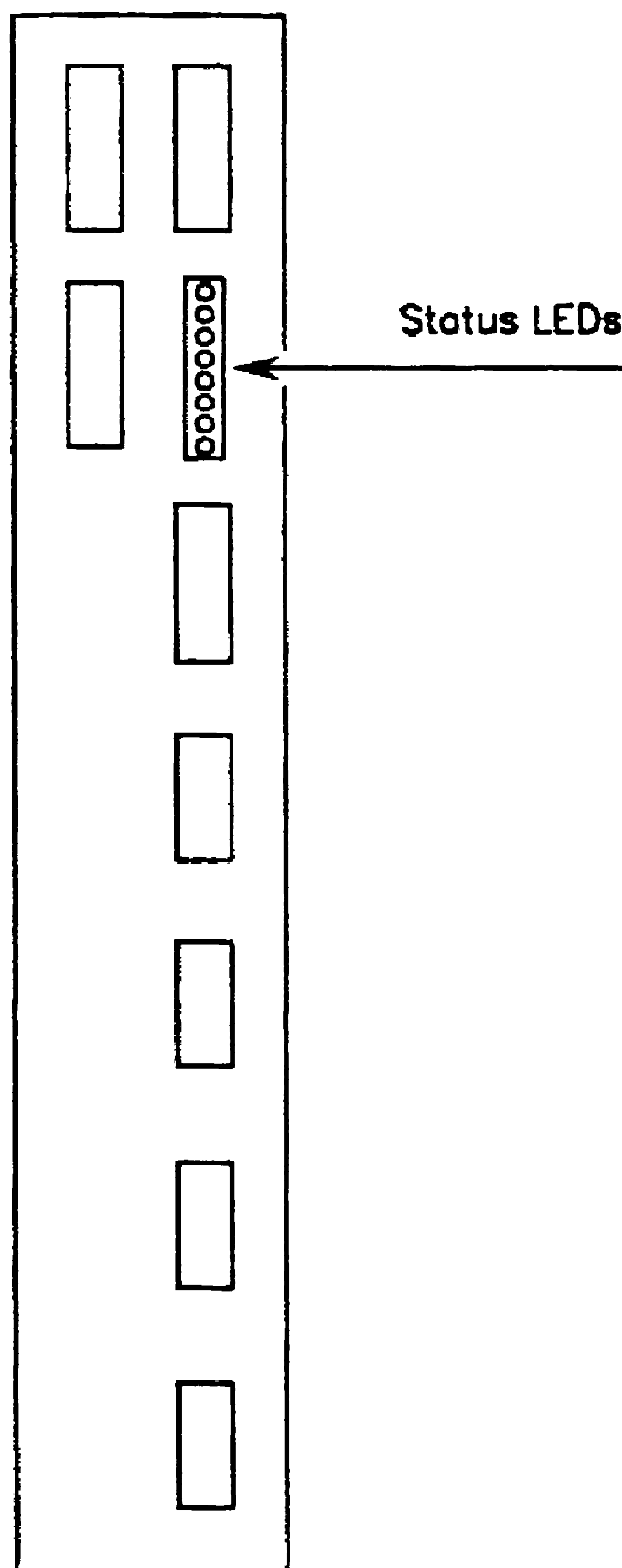
The different status displays for real-mode and protected-mode processors are described below.

**Real-Mode Processors**

The status display on a real-mode processor consists of eight LEDs, as pictured in Figure 19-2. When the processor is functioning normally, the bottom light flashes on and off in a pulsating “heartbeat-like” pattern.

When a processor crashes, the status display enters a loop, during which the error code is displayed by a sequence of LED patterns. The beginning of each loop is heralded by a “walking” pattern. When the LEDs are walking, one light at a time flashes briefly, in sequence from top to bottom (this creates a recycling type of pattern). The next four patterns display the error code. Each pattern is preceded by the top LED blinking off, and then on again. Each pattern corresponds to a hexadecimal digit, which is formed by the LEDs.

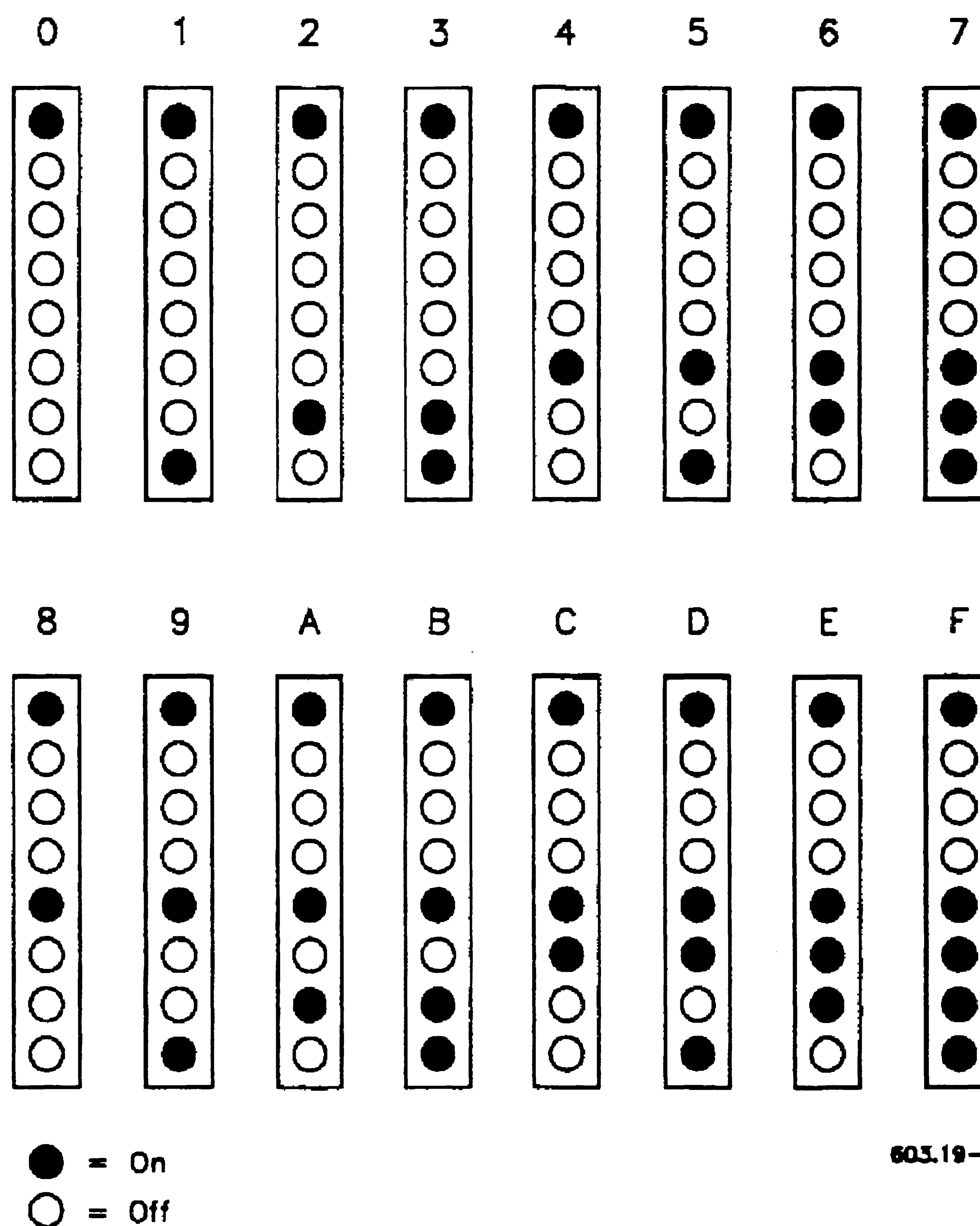
**Figure 19-2. SRP Real-Mode Processor Status LEDs**



603.19-2

Figure 19-3 illustrates the LED pattern for each hexadecimal digit. It is not necessary to understand the hexadecimal (base 16) number system to read and interpret error codes; simply refer to Figure 19-3 to match the pattern to a digit. For a technical explanation of how these digits are derived, see the *CTOS Status Codes Reference Manual*.

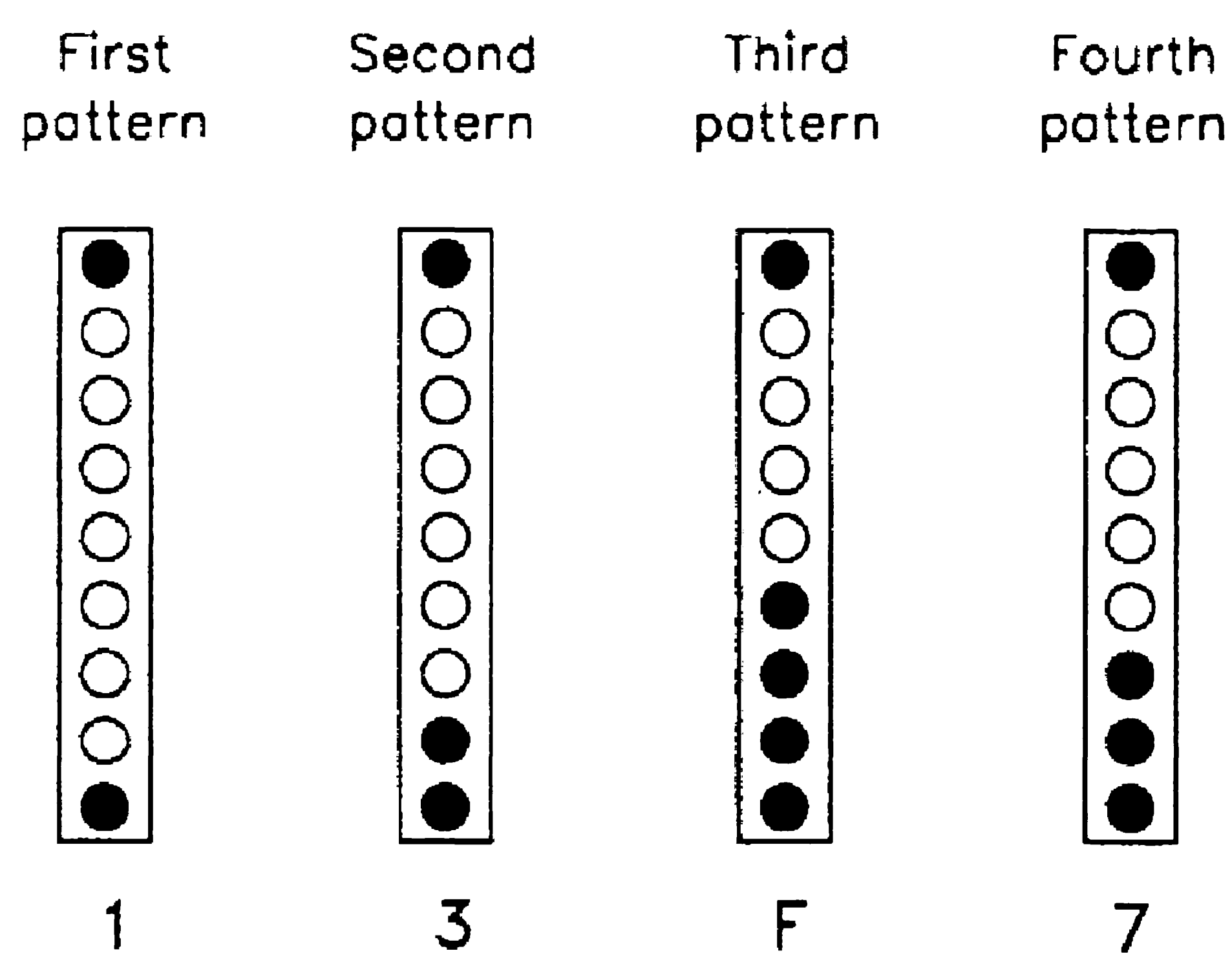
**Figure 19-3. LED Patterns for Hexadecimal Digits**



603.19-3

A multidigit hexadecimal number is then created from the patterns displayed by the LEDs, with the digit represented by the first pattern occupying the highest place. For example, the LED patterns pictured in Figure 19-4 are displayed sequentially to form the hexadecimal number 13F7. Later you will convert the hexadecimal number to a decimal error code number.

Figure 19-4. LED Sequence Pattern for a Hexadecimal Error Code



603.19-4

To interpret a crash error code, follow these steps:

1. Wait for the walking pattern, which signals the beginning of the display loop.
2. Record the hexadecimal number as displayed by the four subsequent LED patterns. If you need to (which is not unusual), copy the LED patterns themselves and refer to Figure 19-3 later. Observe the loop as many times as is necessary to record accurate information.
3. Convert the hexadecimal number to a decimal number (see "Converting Hexadecimal Error Codes," later in this section). The decimal number is the actual error code.
4. Look up the error code (the decimal number) with the **Status Codes Reporter** command.

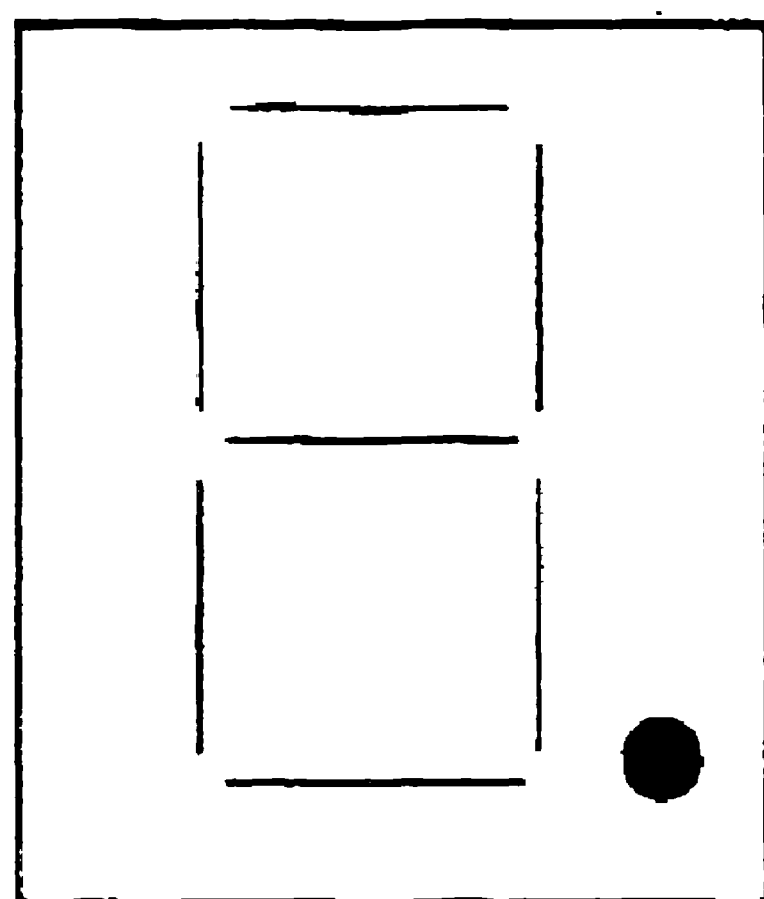
## Protected-Mode Processors

The status display on a protected-mode processor is a seven-segment LED that displays a single digit, as shown in Figure 19-5. A pulsating decimal point indicates that the processor is functioning normally.

When a processor crashes, the status display enters a loop, during which the error code is displayed by a sequence of decimal digits. The beginning of each loop is heralded by a circular pattern around the perimeter of the LED display. The next five digits display the error code.

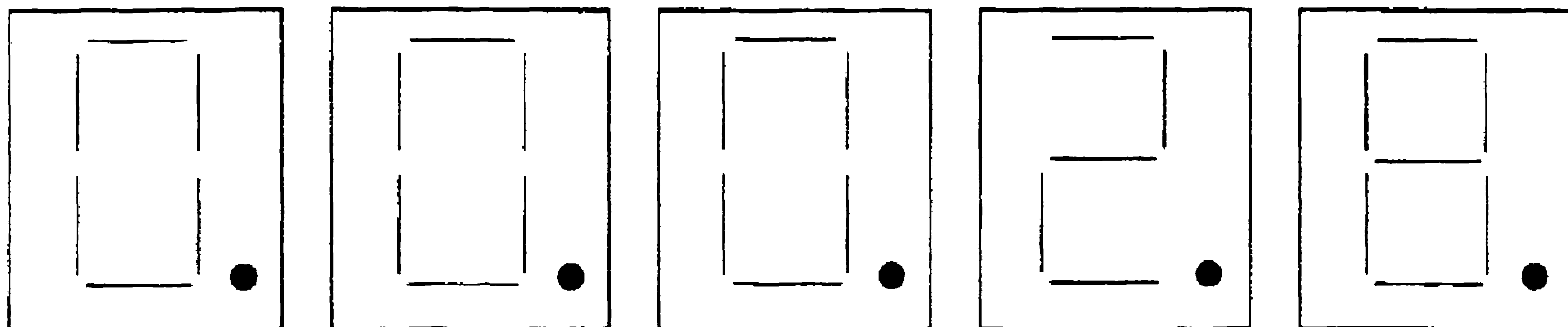
The first digit corresponds to the highest place of the number, for example, error code 28 is displayed as five sequential digits, as shown in Figure 19-6.

Figure 19-5. SRP Protected-Mode Processor Status LED



603.19-5

Figure 19-6. Sequence Pattern for a Decimal Error Code



603.19-6



To interpret a crash error code, follow these steps:

1. Wait for the circular pattern around the perimeter of the LED display.
2. Record each digit that is displayed after the looping pattern. The first digit displayed corresponds to the highest place of the error code number, as described above.
3. Look up the error code with the **Status Codes Reporter** command. On protected-mode processors, error codes are already displayed as decimal numbers, so no conversion is necessary.

***Note:** If a General Processor crashes while it is booting (before the front panel reaches 20), its crash status is displayed in hexadecimal. See Figure 19-7 for a hexadecimal-to-decimal conversion chart.*

### Basic Procedures for SRP Processor Troubleshooting

When a processor board crashes, it usually provides information that can help you determine the cause of the crash. There are several approaches you can take in obtaining and interpreting crash analysis information; the approach you choose in a particular situation will depend on several things, including the kind of work you were doing when the crash occurred, your past experience with this sort of crash, and so forth.

The procedures described below represent the basic approaches to troubleshooting processor crashes and are listed in the recommended sequence. Several procedures are described in more detail later in this section.

1. Read the front panel status display.

The front panel status display provides only the most rudimentary and general description of system problems and is therefore of limited value in troubleshooting. If, for example, a board crash has set the front panel display, this alerts you to the crash if you do not already know about it.

2. Read the processor LED displays.

The crash code on the rear processor display provides more specific information about the crash. It should always be checked when a processor has crashed. Identify the code before you reboot the system.

3. Attempt to reboot the system.
4. After the system reboots, use the **PLog** command to look for crash information in the system error log (see “Diagnosing Problems,” earlier in this section).

As described earlier, the system attempts to log information relating to the error condition. If the error code is successfully written to the log, you can retrieve this information, along with a brief interpretive message, by using the **PLog** command. Note that the error is not always successfully logged, and you should therefore identify the crash code, as described in step 2, before you reset the system and run **PLog**.

5. Call Technical Support if you are unable to isolate and correct the problem.
6. If required, collect a crash dump and copy it to a QIC tape for analysis by a Technical Support engineer (see “Collecting a Crash Dump,” later in this section).

### SRP Processor Crashes During Bootstrapping

Some possible causes of a processor crash during the bootstrap sequence are listed below:

- Corrupted system software
- Corrupted system disk
- Malfunctioning processor board

Try booting the system from a QIC tape. If you are successful, copy the operating system for the processor from the tape to the system volume. This would also be a good time to back up the system volume. If you continue to experience problems, reinitialize the system volume, and then restore the backup.

If the SRP will not boot from tape, follow these steps:

1. Turn off power to the expansion cabinets.
2. Remove all processors from the primary cabinet except the master processor and the processor that keeps crashing (move it to a slot adjacent to the master processor).

3. Disconnect all disk drives from the primary cabinet. (See the installation guide for your SRP.)
4. Attempt to reboot from the QIC tape.

If the processor still crashes while it is booting, you are most likely experiencing a hardware problem with that processor.

### Isolating Hardware Problems

To isolate a hardware problem, you reduce the system to the minimum of hardware required to allow a successful bootstrap. Such a technique is recommended when the system is configured in such a way that the symptoms point to a part that may not be removed without causing additional problems.

If the master processor is a real-mode board, a minimum configuration consists of the master processor and one Cluster Processor. If the master processor is a protected-mode board, a minimum configuration consists of the master processor only. The system is then rebuilt using the original boards and drives until the problem is recreated. This suggests that the part most recently introduced to the system is causing the failure.

An alternate technique requires you to remove hardware, one piece at a time, until the problem cannot be reproduced. This technique is recommended when the system is configured in such a way that the symptoms point to a part that may be removed without causing additional problems. This technique should be used whenever possible to keep the handling of hardware to a minimum and to avoid software reconfiguration.

The system is reduced board by board or drive by drive until the problem no longer exists. The system is then rebuilt using the original boards and drives until the problem is recreated only by the introduction of one remaining part (board or drive), indicating that the part reintroduced to the system is the cause of the original failure.

### Isolating Software Problems

This technique requires you to create the following keyswitch files which bootstrap the system with the minimum of required software:

```
[Sys]<Sys>SrpConfig.k.sys
```

```
[Sys]<Sys>SysInit.k.jcl
```

where *k* is the keyswitch position (*M* for manual, *N* for normal, *R* for remote).

Many system administrators use the *R* (remote) keyswitch position for a minimum software configuration. *SrpConfig.R.sys* usually contains only the configuration parameters required to bootstrap a minimum hardware configuration, and *SysInit.R.jcl* is usually empty. In some situations, however, you may want to install QIC tape system services.

### Errors During System Service Installation

After all processors are booted, the front panel displays 20. At this point, system services are installed from the system initialization file.

Error 50 or 51 on the front panel indicates an error during system initialization. To troubleshoot these errors, bootstrap the system from your troubleshooting keyswitch position or a QIC tape (the purpose being to install a minimum of system services). Then, review and correct the system initialization JCL file for the problematic keyswitch position.

In some cases, the front panel continues to display 20, but a processor “hangs” or appears frozen. This indicates that a system service could not be installed and no further processing can take place because the JCL file is, in effect, still running on the processor.

A processor also hangs if the Remote User Manager (RUM) is not installed with the RunNoWait JCL statement, or if RUM is not the last system service installed on the master processor. See “Cluster View,” in Section 4, “Using Administrative Tools,” and “Installing the Remote User Manager,” in Section 9, “Installing System Services.”

If you suspect such a condition, use **Partition Status** to determine what is running on the processor. If a system service run file is active, the processor is locked.



If a system hangs during system initialization, follow these steps:

1. Reboot a limited software configuration on the SRP.
2. Attempt to install system services manually one at a time, via Cluster View. (See “Installing System Services on an SRP,” in Section 9, “Installing System Services.”)

This allows you to check parameter values and to determine the point at which the installation hangs.

As an alternative to installing system services manually, you can insert FrontPanel statements in the system initialization JCL file. With these statements, you can display a front panel status code at chosen intervals during system initialization. See Table 9-2.

You might need to allocate additional W-, X-, Y-, or Z-blocks if you are installing many system services. See “Adjusting Memory Blocks,” in Section 15, “Optimizing System Performance,” and the parameters for W-, X-, Y-, and Z-blocks, in Section 16, “Configuring Operating Systems.”

## Intermittent System Crashes

Intermittent system crashes can be caused by anything from inadequate power to faulty hardware to mixed revision levels of hardware and software. Use the troubleshooting techniques described earlier in this section to isolate the source of the problem.

If the system stops functioning and no errors are logged, you might be using a power supply that is not adequate for your hardware. Check its revision level and consider upgrading power supplies to the latest models.

## Collecting a Crash Dump

When a processor crashes, a *crash dump* of memory is initiated. During a crash dump, the contents of the processor’s memory at the time of the crash are written to a *crash dump file*. That file can then be analyzed by a system engineer to identify and correct the source of the crash.

As a system administrator, you may be responsible for collecting crash dumps. This section describes the procedures for collecting crash dumps on workstation and SRPs. Information about analyzing crash dump files is provided in the *CTOS Debugger User’s Guide*.

A file named `<Sys>CrashDump.sys` is created when a disk is initialized, and its size is taken from the specified format template (see the `:CrashFileSize:` parameter, in Section 11, “Adding Hard Disks”). Crash dumps are written to the first recognized device containing a crash dump file that is larger than 10 sectors.

*Note:* Access to a file named `<Sys>CrashDump.sys` that is larger than 10 sectors in size is required for a workstation to be bootable.

### Performing Crash Dumps on a Workstation

On workstations, a crash dump takes place automatically when a processor crashes. While the dump is executing, a *D* followed by a series of dots (as when the workstation bootstraps) appear on the screen.

The following subsections describe the crash dump procedures and requirements on the various models of workstations.

#### X-Bus Workstations

On real-mode X-Bus workstations, the crash dump is completed in a single stage and is written to the first recognized device containing a crash dump file that is larger than 0 sectors (in most cases, `[Sys]<Sys>CrashDump.sys`).

On protected-mode X-Bus workstations, an extended crash dump is required to collect the contents of extended memory. After the initial crash dump takes place (as described above for real-mode workstations) the processor reboots using only the first megabyte of memory. Then, an extended crash dump takes place.

If the crash dump file is large enough, the extended crash dump is written to it. If `CrashDump.sys` is not large enough, the extended crash dump is written to an extended crash dump file. The name of the extended crash dump file (usually `[Sys]<Sys>ExtCrashDump.sys`) is specified in the operating system configuration file. See “Creating an Extended Crash Dump File,” later in this section, and the `:ExtCrashDumpFile:` parameter, in Section 16, “Configuring Operating Systems.”

By default, video is deactivated during extended crash dumping. On some workstations, however, that option is configurable; see the `:ExtCrashVDMFile:` parameter, in Section 16, “Configuring Operating Systems.”



## X-Bus+ Workstations

On X-Bus+ workstations, the crash dump is written to the crash dump file and, if that file is large enough, is completed in a single stage. If the crash dump file is not large enough to collect a complete memory dump, an extended crash dump takes place to dump the remaining memory, as described for protected-mode processors, earlier in this section.

The minimum size for the crash dump file on an X-Bus+ workstation is 8192 sectors.

## Diskless Workstations

Diskless workstations dump to *[Sys]<Sys>WS>CrashDump.sys* on the server. To perform extended crash dumping on a diskless workstation, that file specification must be specified as the parameter value for *:CrashDumpFile:* in the operating system configuration file for the diskless workstation; see Section 16, “Configuring Operating Systems.”

## ISA Workstations and Single-Processor EISA Workstations

On ISA workstations and single processor EISA workstations, the entire memory dump is written to *[Sys]<Sys>CrashDump.sys*. To collect a crash dump on an EISA/ISA workstation, *[Sys]<Sys>CrashDump.sys* must be large enough to contain the entire contents of memory. To determine the correct size for that file, in sectors, multiply the size of system memory (in K bytes) by 2. For example, if the system contains 8192K bytes of memory, the correct size for the crash dump file is 16384 sectors.

A crash dump file is created when a disk is initialized, and its size is specified in the format template that is used to initialize the disk. See the *:CrashFileSize:* parameter, in Section 11, “Adding Hard Disks,” for more detailed information.

If memory is added to an EISA/ISA workstation, the crash dump file will no longer be large enough to collect a crash dump. In such a case, the disk must be backed up and then reinitialized with a larger crash dump file.

### Multiprocessor EISA Workstations

Crash dumping on multiprocessor workstations is controlled per processor by the following entry in the operating system configuration file:

```
:Boot:      (Processor = EP01,  
            OS = [Sys]<Sys>vSrvr.img,  
            Dump = Yes)
```

Specify **Yes** to the *Dump* subparameter to implement crash dump collection on the processor.

The master processor dumps to *[Sys]<Sys>CrashDump.sys*, in the same manner described for ISA workstations, above. Each I/O processor card, however, dumps to a unique crash dump file named *[Sys]<Sys>EPnn.crash*, where *nn* is the two-digit processor ID (for example, *EP01.crash*). Those files are created automatically when needed.

If you need to conserve disk space on the system volume, you can specify a path other than *[Sys]<Sys>* for the IOP crash files. See the *:CrashDumpPath:* parameter, in Section 16, "Configuring Operating Systems."

### Creating an Extended Crash Dump File

On X-Bus and X-Bus+ workstations, the extended crash dump file is identified by the *:ExtCrashDumpFile:* parameter in the operating system configuration file.

**Note:** *Extended crash dumping is not supported on EISA/ISA workstations.*

The extended crash dump file must be large enough to contain the entire contents of memory. To determine the correct size, in sectors, multiply the total amount of processor memory by 2. For example, if a processor has 4096K bytes of memory, the crash dump file must be 8192 sectors long. The extended crash dump file can be accessed from any volume.

To create an extended crash dump file, use the **Create File** command, as described in the following procedure.

1. On the Executive command line, type **Create File**; then press **RETURN**.
2. Fill in the command form, as shown in the following example:

```

Create File
File name           [Sys]<Sys>ExtCrashDump.sys
[Vol or dir password]
[File password]
[File protection level] 15
[Size in sectors (0)] 8192
[Overwrite ok?]
  
```

3. Press **GO**.

## Suppressing Crash Dumping

Automatic extended crash dumping is controlled by the following parameter in the operating system configuration file:

*:SuppressAutoDump:*

Set this value to **No** (the default value) to implement automatic extended crash dumping.

If a system is not set up for automatic extended crash dumping, you can perform it manually; see the **Extended Crash Dump** command in the *CTOS Executive Reference Manual*.

## Performing Crash Dumps on an SRP

Crash dumping on SRPs is controlled per processor by the following configuration file entry in *[Sys]<Sys>SrpConfig.sys*:

```

:Boot:           (Processor = CP00,
                  OS = [Sys]<Sys>rSrpCp.sys,
                  Dump = Yes)
  
```

After the *Dump* subparameter, specify **Yes** to implement crash dump collection on that processor.

### Master Processor

On an SRP, the master processor dumps in the same manner described for X-Bus processors. If it is a protected mode processor, up to 4M bytes of memory are dumped to the crash dump file. The processor then reboots in 4M bytes of memory and an extended crash dump takes place. If `[Sys]<Sys>CrashDump.sys` is large enough to accept the entire memory dump (for example, 32768 sectors for a 16M byte processor), the dump of extended memory is appended to that file. If `CrashDump.sys` is not large enough, `[Sys]<Sys>ExtCrashDump.sys` is created and expanded as needed.

After the extended crash dump takes place, the master processor initiates dumping of the other processors and, when that is complete, automatically reboots the system.

### Other SRP Processors

Each of the other processors dumps to a unique file specification. In the default or individual processor section of the operating system configuration file, a volume and directory for the crash dump file is designated, as shown in the following example:

```
:Processor: GP00
:CrashDumpPath: [Sys]<Dump>
```

The specified volume must be controlled by the master processor, and the directory must already exist. Crash dumps files are named `Xpnn.crash`, where `xPnn` is the four-character processor ID, for example, `GP01.crash`.

See the `:CrashDumpPath:` parameter, in Section 16, “Configuring Operating Systems,” for more information.

## What If a System Will Not Bootstrap?

Bootstrapping problems can be caused by the following:

- The system volume is corrupted.
- The operating system in `[Sys]<Sys>SysImage.sys` is not compatible with the workstation or SRP hardware.
- A customized operating system or configuration file contains invalid parameters.



Other causes can be the result of a variety of software and hardware problems. For example, on EISA/ISA workstations, hard disk geometry entries may become corrupted due to loss of battery power. Always check the appropriate hardware manual for additional troubleshooting possibilities.

To troubleshoot a bootstrapping problem, you can attempt to bootstrap from the server or from removable media. If the system still does not boot, you are most likely dealing with a hardware problem.

If the system bootstraps successfully from the server or from removable media, most likely the system volume has been corrupted or an incompatible operating system has been installed.

### Bootstrapping From the Server

If you are troubleshooting a client workstation connected to a server, you can bootstrap it from the server. This is a convenient technique for troubleshooting, because most of the software you need is already installed on the server.

For example, if you need to restore the system volume on a client workstation, simply bootstrap from the server and restore the floppy disks or tape to the client workstation's disk. When you do this, however, remember to use the volume or device name of the disk to which you are restoring (for example, *[d0]* or *[d1]*); do not use *[Sys]*. See "Bootstrapping a Workstation From a Server," in Section 5, "Bootstrapping," for information about bootstrapping from the server.

#### Caution

Remember that when a workstation is booted from the server, *[Sys]* is the server's own system volume. Therefore, when you restore to a disk on the client workstation, designate the volume or device name (such as *[d0]*), rather than *[Sys]*. Restoring to *[Sys]*, while booted from the server, overwrites the server's system volume.

### Bootstrapping From a Floppy Disk

Standard Software includes a utility that allows you to create a set of floppy disks from which you can bootstrap a workstation. You would most likely use this method on a workstation that is not connected to a server or on a workstation that is itself a server. In addition to booting, the floppy disk set allows you to perform the following tasks:

- Sign on to the workstation
- Back up disks
- Restore a backup
- Initialize a disk
- Change the name of a volume

This section contains information and procedures for creating and using a bootable floppy disk set.

### Regular and High-Capacity Drives

You can create a bootable floppy disk set on either a 5.25-inch or a 3.5-inch floppy drive. 3.5-inch drives are high-capacity drives. In addition, 5.25-inch floppy drives in B39 workstations and in MF1, MS5, and MS6 disk modules are high-capacity drives. 5.25-inch floppy drives in other workstations are regular-capacity drives.

If you are not sure whether your drive is high capacity, a 5.25-inch drive will default to regular capacity. Similarly, if you want to create a set of regular capacity bootable floppy disks (so that you can use it on workstations equipped with a regular capacity drive), you can do so on a high-capacity drive by using default parameters.

On 3.5-inch drives, however, you must specify high capacity. If you use default parameters, the floppy disks will not be formatted and the procedure will fail.

Both default and high capacity parameters are described in the procedure for creating the bootable floppy disk set, later in this section.



### Creating a Message File

Before you create the bootable floppy disk set, you need to create a small Executive message file, as described in the following procedure:

1. Generate a message text file from the binary message file named *[Sys]<Sys>ExecMsg.bin*. (See Section 18, "Customizing Standard Software," for detailed instructions.)
2. With the Editor application, open the message text file you generated.
3. Select (highlight) messages 32907 through 32957.
4. While highlighted text is displayed, open a new file named *[Sys]<Sys>CBF>ExecMsg.txt*.
5. Press **COPY** to copy the highlighted text into the new file.
6. Press **FINISH**, then **GO**, to exit the Editor and save the file.
7. On the Executive command line, type **Create Message File**; then press **RETURN**.
8. Fill in the command form as shown below:

Create Message File

Text file                    [Sys]<Sys>CBF>ExecMsg.txt

[Message file]             \_\_\_\_\_

[Print file]                \_\_\_\_\_

9. Press **GO** to create *[Sys]<Sys>CBF>ExecMsg.bin*.

### Creating a Bootable Floppy Disk Set

Media requirements for the bootable floppy disk set are three floppy disks for a high-capacity drive, or five floppy disks for a regular-capacity drive.

To create the bootable floppy disk set, follow these steps:

1. On the Executive command line, type **Create Backup Floppy**; then press **RETURN**.
2. If required, fill in the command form, as shown in the following example. Parameter fields are described in Table 19-1.

```
Create Backup Floppy
[Include tape utilities?]  Yes
[High capacity floppies?]
[Floppy drive (f0)]       [f1]
```

**Note:** *Because **Create Backup Floppy** is executed from a submit file, any response in a field activates the parameter. Therefore, to respond **No** to either field, leave it blank.*

3. Press **GO**.

After some introductory messages are displayed, you are prompted to insert the first floppy disk. Instructions appear on the screen; continue responding to messages and changing disks as prompted.

**Table 19-1. Create Backup Floppy Parameters**

Field Name	Description
<i>[Include tape utilities?]</i>	<p>Default: No</p> <p>Specify <b>Yes</b> if you want the Sequential Access Service utilities to be included on the floppy disk set. If you do not want to include the tape utilities, leave this field blank.</p>
<i>[High capacity floppies?]</i>	<p>Default: No</p> <p>The following possibilities exist for this field:</p> <ul style="list-style-type: none"> <li>• If the workstation is equipped with a 3.5-inch drive, specify <b>Yes</b>. (If you leave this field blank, the command will fail.)</li> <li>• If the workstation is equipped with a regular-capacity 5.25-inch drive, leave this field blank. (If you specify anything in this field, the command will fail.)</li> <li>• If the workstation is equipped with a high-capacity 5.25-inch drive, and you want to create a high-capacity floppy disk set, specify <b>Yes</b>.</li> <li>• If the workstation is equipped with a high-capacity 5.25-inch drive and you do not want a high-capacity floppy disk set, leave this field blank.</li> </ul> <p>See also "Regular and High-Capacity Drives," earlier in this section.</p>
<i>[Floppy drive (f0)]</i>	<p>Default: <i>[f0]</i></p> <p>Enter the device specification of the floppy drive to be used to create the bootable floppy disk set.</p>

### Labeling the Bootable Floppy Disk Set

As the floppy disks are created, label them sequentially, as follows:

#### Regular-Capacity Set

Boot Disk

Continuation Disk 1: Optional Tape Services (You will need to use this disk even if you did not include tape services.)

Continuation Disk 2: Volume Archive, Change Volume Name

Continuation Disk 3: Format Disk

Continuation Disk 4: Restore Archive

#### High-Capacity Set

Character or Bitmap Boot Disk

VGA Boot Disk

Continuation Disk

### Using the Bootable Floppy Disk Set

To bootstrap a workstation from the floppy disk set, follow these steps:

1. Turn on the workstation.
2. Insert the Boot Disk into the floppy disk drive.
3. Press the Reset button.

When the workstation has successfully booted, the following message appears:

```
    Please insert Continuation 1 Diskette
          Press any key to continue
```

4. Remove the Boot Disk.
5. Insert the Continuation 1 Disk.
6. Press **GO**.
7. Insert the appropriate continuation disk, as described on the screen.
8. Press **GO**.

With the high-capacity floppy disk set, you do not need to change disks between backup, format, and restore procedures. With the regular-capacity floppy disk set, however, you need to change disks. Be sure to wait for commands to finish executing before changing to a different floppy disk.

In addition to commands listed on the continuation diskettes, the following commands are available: **Batch Foreground, Copy, Create Directory, Create File, Delete, List, Logout, Path, Rename, Run, and Type.**

## Bootstrapping From QIC Tape

To troubleshoot an SRP, you can bootstrap from the CTOS/XE Boot Tape, which is supplied with the distribution media. The boot tape creates a limited, but valid, system volume in memory on the SRP.

### Using the Bootable Tape

To boot an SRP from tape, the SRP must contain one of the following minimum hardware configurations:

- One protected-mode processor
- Three real-mode processors

(This is the same configuration that is required for you to install the CTOS/XE system software.)

In addition to the SRP hardware requirement, one client workstation that boots locally must be connected to the SRP as follows:

- If the master processor is a General Processor with SCSI Interface, connect the client workstation to a cluster port on the master processor.
- If the master processor is a File Processor or Data Processor, connect the client workstation to channel A on the first Cluster Processor.

You will use this workstation to communicate with the SRP after it boots from tape; therefore, make sure that the Cluster View commands are installed on the workstation.

Because only a minimum number of processors boot from the tape, it is essential to connect the client workstation to the correct cluster channel. To simplify troubleshooting, many system administrators connect a workstation, in close proximity to the SRP, to use specifically as a troubleshooting station.

To bootstrap from the CTOS/XE Boot Tape, follow these steps:

1. Insert the tape into the QIC tape drive on the SRP.
2. Turn off the keyswitch, and then turn it back on to its usual position.

When the front panel reads 20, the SRP has bootstrapped a minimum hardware and software configuration. You have access, via Cluster View, to internal Executive commands, as well as the following commands: **Format Disk**, **Install Sequential Access Service**, **Restore Archive**, **Submit**, and **Volume Archive**.

You can execute other commands, such as **Volume Status**, **Cluster Status**, **Files**, and **Editor**, on the client workstation.

### Creating the Bootable Tape

If you use a customized operating system on your SRP, you may want to create your own bootable tape for troubleshooting. In addition, if your SRP exceeds the minimum hardware requirements, you can add more files to those contained in the system volume created in memory. This expands the troubleshooting tools available to you when you boot from tape.

See the **Create Boot Tape** command in the *CTOS Executive Reference Manual* for information about creating a bootable tape.

As a safeguard for your master copy of the CTOS/XE Boot Tape, you can make a copy of it to use for troubleshooting. To do so, use the **Tape Copy** command to copy tape files 0 through 7. See the *CTOS Executive Reference Manual* and the release documentation for the CTOS/XE operating systems.



## Error 21 on the Front Panel

Error 21 on the front panel means that the bootstrap ROM cannot locate a bootable file named <Sys>*SysImage.sys*. If the front panel displays 21 when you attempt to boot from QIC tape, suspect a faulty disk-drive cable or a hardware problem on the master processor.

## Converting Hexadecimal Error Codes

In most cases, you will need to convert hexadecimal error codes into decimal numbers before you can look them up in the *CTOS Status Codes Reference Manual*. The chart shown in Figure 19-7 will help you perform that task.

Each hexadecimal error code consists of four digits, and as with decimal numbers, the leftmost digit holds the highest place value. For example, in the hexadecimal number 13F7, the digit 1 holds the highest place, 7, the lowest.

Figure 19-7 is arranged in four columns with each column corresponding to one digit of the hexadecimal error code. It is arranged from highest to lowest digit, starting from the left, in the same order as the hexadecimal number. Along the left of each column is the hexadecimal digit itself; along the right is a corresponding decimal value.

To use Figure 19-7 to obtain the decimal error code number, follow these steps:

1. Locate the hexadecimal digit for each place value and note the corresponding decimal equivalent for each.
2. Add all the decimal equivalents together to obtain the decimal error code number.

For example, the decimal equivalents for the hexadecimal number 13F7 are as follows:

Hexadecimal Digit	Decimal Equivalent
1	4096
3	768
F	240
7	7

The total, 5111, is the decimal error code number.

Figure 19-7. Hexadecimal-to-Decimal Conversion Chart

Highest Digit <span style="float: right;">→</span> Lowest Digit							
HEX	DEC	HEX	DEC	HEX	DEC	HEX	DEC
0	0	0	0	0	0	0	0
1	4,096	1	256	1	16	1	1
2	8,192	2	512	2	32	2	2
3	12,288	3	768	3	48	3	3
4	16,384	4	1,024	4	64	4	4
5	20,480	5	1,280	5	80	5	5
6	24,576	6	1,536	6	96	6	6
7	28,672	7	1,792	7	112	7	7
8	32,768	8	2,048	8	128	8	8
9	36,864	9	2,304	9	144	9	9
A	40,960	A	2,560	A	160	A	10
B	45,056	B	2,816	B	176	B	11
C	49,152	C	3,072	C	192	C	12
D	53,248	D	3,328	D	208	D	13
E	57,344	E	3,584	E	224	E	14
F	61,440	F	3,840	F	240	F	15

603.19-7

# Appendix A

## Workstation Bootstrap and Initialization Codes

### Introduction

When the workstation starts up it goes through a boot process generated by firmware.

On X-Bus workstations the boot ROM firmware initializes the hardware, then loads the operating system. If it finds any problems before it loads the operating system it shows an error code on the screen or on the keyboard LEDs.

On EISA or ISA workstations the BIOS firmware initializes the hardware. If no ClientCard is present the normal boot sequence is to check the floppy disk drive for a bootable system image. If no floppy disk is found in the drive then the hard disk drive is checked for a bootable system image. Any error messages generated in this process are from the BIOS. If a ClientCard is present, it intervenes in this process to provide booting from the server as an additional option. ClientCards generate some messages that are the same as those for X-Bus workstations. An additional set of errors are also generated on EISA and ISA workstations if errors are found when trying to load `[Sys]<Sys>Sysimage.sys`.

Once the firmware loads the system image, control is passed to the operating system, which goes through its own initialization processes. At this time the system configuration file is read and IOP cards are booted. Additional status messages, called *initialization status codes*, may be displayed at this time. Note that any errors encountered in initializing the IOP are also written to the system log file at this time.

## Workstation Bootstrap and Initialization Codes

---

Operating system initialization codes are the same for all workstations. Boot errors can vary with type of workstation. This appendix first describes operating system initialization errors, then lists boot ROM errors in the following order:

- B26, B27, B28, B38, and B39 workstations and the ClientCard
- B24 workstations
- SuperGen EISA and ISA workstations
- SuperGen Series 5000 workstations
- SuperGen SG2100-320 workstations

BIOS errors are described in your hardware installation guide. IOP error messages are described in Appendix D, "CTOS I/O Processor Card Error Codes."

# Operating System Initialization Status Codes

The operating system tests the following hardware components during its initialization process:

- Memory
- Memory parity error detection circuits
- Keyboard
- Interrupt circuits
- Programmable interval timer (PIT)

The video circuits are included in the tested hardware. If the test of these video components fails, nothing can be displayed on the screen, and the operating system halts and beeps 10 times. For status codes 0 to 3, the operating system also displays the code on I-bus keyboard LEDs (**F10**, **F9**, **F8**, and **F3** for codes 0 to 3, respectively).

If the video test succeeds, but any of the other tests fail, the operating system continues to load the Executive, and the error code is displayed as a hexadecimal number.

Typical initialization errors are:

80h	Keyboard not plugged in
30h	Hardware memory failure
400h	Bad keyboard or typing during initialization

## Initialization Codes

The error conditions numbered 0 to F (hexadecimal) are defined below:

- 0** LED **F10** is lighted. This means that the video hardware does not respond. Possible causes are listed below:
- There is no video board.
  - The video board is not seated.
  - The cables are loose or unconnected.
- 1** LED **F9** is lighted, indicating a DMA failure in “load font.”

## Operating System Initialization Status Codes

---

- 2** LED **F8** is lighted, indicating a DMA failure in “read font.”
- 3** LED **F3** is lighted. The font read back from the Font RAM fails to compare with the font written to it.
- 4** Memory test failure.
- 5** Bus timeout interrupt is not generated when a nonexistent memory location is referenced.
- 6** Invalid memory parity is not detected.
- 7** Keyboard hardware does not respond. Possible causes are listed below:
- The keyboard is not connected.
  - The keyboard hardware is faulty.
- 8** Keyboard does not return **GOOD** status after the reset command. Possible causes are
- Keys were pressed during the initialization.
  - The keyboard hardware is faulty.
- 9** Keyboard ROM checksum failure.
- A** Keyboard loopback test failure.
- B** Keyboard interrupt test failure.
- No interrupt is generated during loopback, or **TRANSMIT READY** status in 8251A does not generate interrupt.
- C** Real-time clock (RTC) test failure.
- No interrupt occurred, or the time interval between two RTC interrupts was inconsistent with the time interval measured by programmable interval timer (PIT).
- D** PIT test failure.
- No PIT interrupt.
- E** Continuous PIT interrupts.
- F** Communications hardware test failure (cluster or server workstation only).



## Boot ROM Codes

When you bootstrap a workstation, it runs diagnostic and bootstrapping routines that are resident in the ROM of the CPU. When the boot ROM detects an error, the error code usually appears on the screen as a two-digit hexadecimal code.

For E0h and E1h error codes only, most workstations beep five times and the error code is displayed on the keyboard LEDs. (An E0h error code indicates a bad ROM, and an E1h code indicates a bad RAM.) Tables A-1 and A-2 show how to interpret these status codes on the keyboard and speaker.

The B28-EXP workstation reports some bootstrap codes on the speaker only. The B27 workstation reports the E0h and E1h error codes on both the keyboard LEDs and the speaker. For both systems, a lighted LED corresponds to binary number 1 and an unlighted LED to a binary number 0. On the speaker, a short tone is binary number 1 and a long tone is binary number 0.

**Table A-1. Boot ROM Error Code E0h**

Key	LED	Speaker	Binary
OVERTYPE	on	short	1
LOCK	on	short	1
F1	on	short	1
F2	off	long	0
F3	off	long	0
F8	off	long	0
F9	off	long	0
F10	off	long	0

## Boot ROM Codes

---

Table A-2. Boot ROM Error Code E1h

---

Key	LED	Speaker	Binary
OVERTYPE	on	short	1
LOCK	on	short	1
F1	on	short	1
F2	off	long	0
F3	off	long	0
F8	off	long	0
F9	off	long	0
F10	on	short	1

---

### Boot ROM Codes for X-Bus Workstations and the ClientCard

This subsection lists boot ROM error codes for B26, B27, B28, B38, B39, and SG2100-320 workstations. It also lists error codes that may be reported by the ClientCard. The codes are given in hexadecimal format.

See “Boot Codes for EISA and ISA Workstations,” later in this section for additional information on errors that may occur on workstations that use the ClientCard. See “Boot Codes for SG2100-320” later for additional codes that may be reported by that workstation.

00

to

01

Reserved.

02

No floppy controller.

The processor cannot communicate with floppy controller. Check the following:

- The floppy disk drive module connections to the X-Bus; on a B27, check the dual floppy disk module connection to the F-bus
- The floppy disk module connection to the cables
- The floppy controller

03

Timeout waiting for an interrupt after a Seek operation.

The floppy disk controller did not interrupt the CPU after being issued a Seek operation. Check the following:

- User intervention (floppy disk drive door may have been opened)
- The dual floppy disk module connection to the X-Bus; on a B27, check the dual floppy disk module connection to F-bus

04

Data-in bit set in command.

The floppy controller was trying to send information to the processor while the processor was trying to send a new command to the controller. The code usually means that the floppy disk controller received an invalid command.

**05** Data-in bit not set in result.

When the processor expected result information, none was available. This error is similar to error code 03.

**06** DMA never finished.

The byte count register of the 80186 DMA channel 0 never decreased to 0, which means that the DMA operation never finished. Check the following:

- The CPU board (processor module)
- The floppy controller
- The floppy disk module; on a B27, check the dual floppy disk module

**07** Volume home block checksum error.

The first block of information read into memory from the floppy disk is invalid. This error can occur because:

- The floppy disk was never initialized.
- The floppy disk is defective.
- The DMA logic is storing the wrong data in memory.
- The floppy controller is malfunctioning.

Check the floppy controller and the floppy disk drive.

**08** No file on floppy disk.

The volume home block on the floppy disk gives a length of 0 for *SysImage.sys* and/or *CrashDump.sys*. When the floppy disk was initialized, the number of pages recorded for the file length was zero. Check the following:

- That the floppy disk was initialized by the **Format Disk** command
- That a valid run file was copied to the system image file
- The size of the crash dump file
- The memory board and the memory expansion board
- The CPU board
- The floppy/hard disk module

- 09** Run file checksum error.  
The floppy disk contains no run file.  
The system image file on the floppy disk does not contain a valid run file. The **Format Disk** command does not automatically copy a system image file onto the volume being initialized; therefore, the user must copy a valid system image file onto the volume.
- 0A** File header invalid.  
Check the floppy disk.
- 0B** Floppy control register inconsistent.  
The status register was polled until it became ready. The floppy status register was then polled again and it was not ready. Check the following:
- The floppy drive door or locking mechanism
  - The connections of floppy disk drive
  - The floppy controller
- 0C** Floppy disk drive was not ready while performing a seek.  
This error can be caused by opening the door of the floppy disk drive or by a bad cable from the floppy disk drive to the motherboard. Check the following:
- The floppy drive door or locking mechanism
  - The cable from the floppy disk drive to the floppy controller
  - The floppy disk drive
  - The floppy controller
- 0D** Invalid floppy disk command received.  
This error occurs when the floppy controller receives an undefined command during the seek/recalibrate phase. It is usually caused by a defective floppy controller.
- 0E** Floppy disk drive not ready.  
The floppy disk drive was not ready when the seek or recalibrate operation was performed. Check whether the floppy disk is inserted properly.

**0F** Floppy disk drive fault condition during a seek or recalibrate operation.

The floppy disk drive did not recalibrate to track 00 after 77 step pulses, or the drive fault line went active. Check the following:

- The floppy drive door or locking mechanism
- The cable between the floppy disk drive and the floppy disk controller board
- The floppy disk and drive
- The floppy controller

**10** Abnormal termination of a seek operation.

The floppy disk drive did not complete the seek operation correctly. Either the floppy disk drive failed or the ready status changed. Check the following:

- The floppy disk
- The cable between floppy disk drive and floppy controller
- The floppy disk drive
- The floppy controller

**11** Floppy disk drive not ready.

The floppy disk drive was not ready when a read or write operation was performed. This error can occur only if the floppy disk drive was previously ready during a recalibrate and a seek operation. Check the following:

- User intervention (the floppy disk may have been removed)
- The floppy disk for damage
- The cable between the floppy disk and the floppy controller board
- The floppy disk drive
- The floppy controller

**12** Invalid floppy disk command received.

The floppy controller received an undefined command when the ROM requested a data transfer.



**13** Floppy disk drive is not ready.

The floppy disk drive was not ready when a read or write operation was performed. This error can occur only if the floppy drive was previously ready during a recalibrate and a seek operation.

Check that the disk was not removed.

B27: Drive is busy.

The floppy disk drive was busy when a read or write operation was performed. Check the following:

- User intervention
- The floppy disk
- The cable between the floppy disk and the floppy controller board
- The floppy disk drive
- The floppy disk controller

**14** Reserved.

**15** End of track.

After a read or write operation, no terminal count signal was received from the DMA. Check the following:

- The cable between the floppy disk drive and floppy controller
- The CPU board (processor module)
- The floppy controller

**16** Data error (data field).

The floppy controller is unable to read data from the floppy disk drive correctly. Check the following:

- The floppy disk
- The cable between the floppy disk drive and floppy controller board
- The floppy disk drive
- The floppy controller

**17** Data error (identification field).

The floppy controller is unable to read the identification field of the addressed sector. Check the following:

- The floppy disk
- The cable between the floppy drive and floppy controller
- The floppy drive
- The floppy controller

**18** Data late.

The floppy controller did not get service from the DMA in time. Check the following:

- Seating of the CPU on the motherboard
- The CPU board (processor module)
- The floppy controller
- The floppy disk module connection to the X-Bus; on a B27, check the dual floppy disk module connection to the F-bus

**19** No data: wrong cylinder.

During a read or write operation, the floppy disk was on the wrong cylinder. This means either that the floppy disk was not initialized correctly or that a seek went to the wrong place. Check the following:

- The floppy disk
- The floppy drive
- The cable between the floppy drive and floppy controller

**1A** No data: bad cylinder.

The cylinder accessed is marked as number 255 (0FFh). Check the following:

- The floppy disk
- The floppy controller
- The connections of floppy disk drive/module

**1B** No data.

The floppy controller reported a no data condition. This means that the specified sector could not be found; most likely, the floppy disk is defective. Check the following:

- The floppy disk
- The floppy drive
- The cable between the floppy drive and floppy controller
- The floppy controller

**1C** Floppy disk is write-protected.

This error code appears only during the dump operation and indicates that the floppy disk has a write-protect tab in place.

Check the floppy disk.

**1D** Missing address mark: data field.

The floppy controller was not able to find the data address mark of a sector. This usually means that the floppy disk is defective.

Check the following:

- The floppy disk
- The floppy disk drive
- The cable between the floppy disk drive and floppy disk controller
- The floppy controller

**1E** Missing address mark: ID field.

The floppy controller was not able to find any ID address mark on a track. This usually means that the floppy disk has not been initialized. Check the following:

- The floppy disk
- The floppy drive
- The cable between the floppy drive and floppy controller
- The floppy controller

## Boot ROM Codes for X-Bus Workstations and the ClientCard

---

- 1F** Abnormal termination of command.  
The floppy controller reported abnormal termination of a command without reporting the cause.  
Check the floppy controller.
- 20** Incompatible processor.  
A disk module is incompatible with the processor. Use a later disk module with this processor.
- 21**  
**to**
- 23** Reserved.
- 23** Timeout waiting for an interrupt.  
The hard disk controller did not interrupt the CPU after being issued a command.  
Check the following:
- The hard disk module connection to the X-Bus; on a B27, check the floppy/hard disk module connection to the F-bus
  - The hard disk controller board
  - The CPU board (processor module)
- 24**  
**to**
- 25** Reserved.
- 26** DMA not finished.  
The word count register of the 8237 DMA channel 3 never decreased to 0FFFFh, which means that the DMA operation never finished.  
Check the following:
- The CPU board (processor module)
  - The hard disk controller
  - The hard disk drive/module connections

- 27** No valid volume home block.  
No volume home block could be found within the first track of the disk.  
Check that the disk was initialized by the **Format Disk** command.
- 28** No file.  
No system image or crash dump file exists.  
Check that the disk was initialized by the **Format Disk** command.
- 29** Run file checksum error.  
The system image file on the hard disk failed a checksum test.  
Check the following:
- That a valid run file was copied to the system image file
  - The memory board and memory expansion cartridges (processor module)
  - The CPU board (processor module)
  - The floppy/hard disk module
- 2A** File header invalid.  
The system image file on the hard disk does not contain a valid run file. Because the **Format Disk** command does not automatically copy a system image onto the volume it is initializing, the user must copy a valid system image onto the volume.  
Check the hard disk system image file.
- 2B** Hard disk status register inconsistent.  
The status register was polled until it became ready. The status register was then polled again and it was not ready. Check the following:
- The hard disk
  - The hard disk drive cables
  - The hard disk controller board

## Boot ROM Codes for X-Bus Workstations and the ClientCard

---

**2C** No seek complete.

The seek-complete status was not set after an implied seek. Check the following:

- The cable from the hard disk drive to the hard disk controller motherboard
- The hard disk drive
- The hard disk controller

**2D**

to

**30** Reserved.

**31** Drive not ready.

The hard disk drive was not ready during an operation. Check the following:

- The cable from the hard disk drive to the hard disk controller
- The hard disk drive, logic, and jumper configuration
- The hard disk controller

**32** Reserved.

**33** Hard disk controller was not ready.

The hard disk controller remained in a busy state after a write or read operation should have completed. Check the following:

- The hard disk controller
- The hard disk drive

**34** Reserved.

**35** Sector not found.

The hard disk controller could not locate a particular sector on a track. Check the following:

- That the hard disk has been formatted
- The cables between the hard disk drive and hard disk controller
- The hard disk controller
- The hard disk drive



- 36** Data error (data field).  
The hard disk controller could not read data from the hard disk drive correctly. Check the following:
- That the hard disk has been formatted
  - The cables between the hard disk drive and hard disk controller
  - The hard disk controller
  - The hard disk drive
- 37** Reserved.
- 38** Data late.  
The hard disk controller did not get service from the 8237 DMA controller in time. Check the following:
- Seating of the CPU board on the processor-module motherboard
  - The CPU board (processor module)
  - The hard disk controller
  - The floppy/hard disk module connection to the X-Bus; on a B27, check the floppy/hard disk module connection to the F-bus
- 39**  
**to**
- 3B** Reserved.
- 3C** Hard disk write fault.  
This error code appears only during a dump operation. Check the following:
- The cables from the hard disk drive to the motherboard
  - The hard disk controller
  - The hard disk drive
- 3D**  
**to**
- 3E** Reserved.

## Boot ROM Codes for X-Bus Workstations and the ClientCard

---

- 3F** Abnormal termination of command.  
The hard disk controller reported abnormal termination of a command without reporting the cause.  
Check the hard disk controller.
- 40** Reserved.
- 41** Unexpected interrupt from the tape controller.  
Check the tape interface board.
- 42** Bad run file signature.  
The run file on the cartridge is not valid. Recopy the run file to the tape with the **Tape Copy** command.
- 43** Tape controller timeout.  
The tape controller is not responding.  
Check the tape interface card and the tape drive.
- 44** Not enough memory.  
The run file being booted from tape is larger than available memory.  
Add more RAM to the workstation.
- 45** Run file checksum error.  
The run file is bad. Recopy the run file to the tape, or try a new tape cartridge.
- 46**  
**to**  
**4F** Reserved.
- 50** No valid volume home block.  
No volume home block could be found within the first track of the SCSI disk device.  
Check that the disk has been initialized.

- 51** No request from target.  
The SCSI device is not asserting the REQ signal. Check the following:
- That the device is connected
  - That the device has been powered up
- 52** No boot or dump file exists on the device.  
Check that the disk has been initialized.
- 53** The volume home block for a disk device contains an invalid block address.
- 54** Invalid run file signature.  
The file being booted is not a valid run file.
- 55** Invalid run file.  
The run file being booted has failed a checksum test.  
Check that the boot device has been properly initialized.
- 56**  
**to**  
**57** A SCSI device is asserting BUSY continuously. The boot ROM cannot obtain the bus.  
Check:
- The X-Bus module latches
  - The internal SCSI cables
  - SCSI devices
- 58** SCSI bus inconsistency.  
The SCSI bus state was sampled twice with inconsistent results. This may be a connection, cabling, or bus termination problem.

- 59** SCSI interrupt timeout.
- No completion interrupt occurred after a SCSI bus operation was started. Check the following:
- The SCSI controller
  - The SCSI device
  - System DMA components
- 5A** Cannot select SCSI target.
- The boot ROM cannot select the SCSI target. Check that the device is connected and is powered up.
- 5B** SCSI target is busy.
- The SCSI device responds with BUSY status following a command. Check the SCSI device.
- 5C** A check condition was received after a request-sense operation. Check the following:
- The SCSI device
  - The SCSI device cables
- 5D** A SCSI status phase was never entered.
- Check the SCSI device for problems.
- 5E** A SCSI message phase was never entered.
- Check the SCSI device for problems.
- 5F** An unknown SCSI phase was detected. Check the following:
- The SCSI device
  - The SCSI device cables
- 60** No active partition.
- Use the **Disk Partition Set Active** command or the MS-DOS **FDisk** command to set an active partition.

- 61** Error loading the operating system.  
A valid partition table exists and an active partition table exists.  
The system image file may be missing or invalid.
- 62** No boot record exists.  
The boot record in an active partition is bad.
- 63**  
**to**  
**6F** Reserved.
- 70**  
**to**  
**A0** Reserved.
- A1** B27: Comm dump wait UA.  
No UA was received to acknowledge a sector sent to the server.  
Check the following:
- The I/O board
  - The server, which may have crashed
  - The communications cable
- A2** Never polled.  
This indicates a nonfatal error and occurs when a disk waits 10 seconds for an initial poll from the server workstation. This error is reported during the boot initialization routine as part of error code 40h. It is never displayed in the LEDs of the keyboard. Check the following:
- The communications cable
  - The I/O memory board
  - The server's power, system files, or whether a **Disable Cluster** command was issued
  - If not clustered, the hard disk system files

### **A3** Serial input/output error.

The serial input/output initialization routine detected an error in the serial input/output communications controller chip. Check the following:

- The I/O board (processor module)
- The CPU board (processor module)
- The cluster controller logic

### **B27:** Comm init SIO error.

The communications line is always busy. The workstation cannot send without causing a collision. Check the following:

- The I/O board
- The server workstation, which may have crashed
- The communications cable

### **A4** 8254 error.

The clock initialization routine detected an error in the 8254 programmable counter/timer chip.

Check the CPU board (processor module).

### **A5** No SIM.

RIM was sent to the server workstation, but no SIM was received. This indicates that the workstation is able to receive but not transmit, or that the server workstation is able to transmit but not receive.

Check:

- The cluster communications logic on the I/O board (processor module)
- The server workstation (including the operating system, which may have crashed)
- The communications cable



**A6** No UP in initialization (SNRM).

A UA or XID was sent to acknowledge the SIM sent by the server workstation, but the server workstation sent back an SNRM instead of a UP. The server workstation probably timed out, while waiting for the UA or XID. Check the following:

- The I/O board (processor module)
- The server workstation (including the operating system, which may have crashed)
- The communications cable

**A7** No UP in initialization (DISC).

A UA or XID was sent to acknowledge the SIM sent by the server workstation. The server workstation sent back a DISC instead of a UP.

Check the following:

- The I/O board (processor module)
- The server workstation, which may have crashed
- The communications cable

**A8** Reserved.

**A9** No identification available.

The initialization routine monitored the cluster communications line but did not find a free identification number. This is usually caused by attaching more workstations to a cluster communications line than the operating system of the server workstation is designed to accept.

Check the operating system of the server workstation, which may have crashed.

**AA** Identification failure.

The initialization routine found free workstation identification numbers by monitoring the communications line, but errors were detected when it tried to use one. This is usually caused by a failure of the collision recovery algorithm. To recover, press the reset button of each workstation that collided, or turn the workstations off and then back on.

**AB** Read identification timeout.

The initialization routine timed out after waiting 10 seconds while monitoring the communications line for a workstation identification number. This error code is generated only after a number of unsuccessful read attempts. Check the following:

- The I/O board (processor module)
- The server workstation, which may have crashed
- The communications cable

**AC** Bad address (dump routine).

The workstation identification number sent in a frame by the server workstation did not match the one expected. Check the following:

- The communications cable, connections, and termination
- The I/O board (processor module)
- The cluster communications logic

**AD** Disconnected (dump routine).

The server workstation sent a DISC because of excessive line or protocol errors or because of a conflict with the crash/dump file at the server workstation. Check the following:

- That either the file *[Sys]<Sys>WSnnn>CrashDump.sys* or *[Sys]<Sys>WS>CrashDump.sys* exists at the server workstation
- That the file is not in use by another workstation that is dumping
- That the crash dump file is large enough

Check also the following:

- The communication cable, connections, and termination
- The I/O board (processor module)
- The cluster communications logic

On an SG2100-320 workstation, this may be a warning only, not an error. These systems can dump all memory if the crash dump file is large enough. This would eliminate the intermediate crash dump step, and no `erc AD` would be received. However, if the system has 6MB of memory and the crash dump file is 2MB (the minimum size), the ROM dumps 2MB, prints the `erc AD`, and reboots. It then runs the extended crash dump to dump the remaining 4 MB, and reboots again.

**AE** No UP - SNRM.

Check the following:

- The I/O board (processor module)
- The server workstation

**AF** No UP - REJ.

Check the following:

- The I/O board (processor module)
- The server workstation

**B0** No UP.

After transmitting a dump block, an unexpected response was received from the server workstation. Check the following:

- Whether a cluster workstation is using the fixed identification mode
- The I/O board (processor module)
- The cluster communications logic
- The cluster cables, connections, and termination

**B1** Read UI error.

A bootstrap block (frame type UI) was expected, but another frame type was received.

Check the I/O board (processor module).

**B2** Read SNRM error.

A bootstrap block (frame type UI) was expected, but a SNRM was received.

Check the I/O board (processor module).

**B3** Disconnected.

The server workstation sent a disconnect because of a conflict with the system image file, or possibly because of excessive errors during transmission. Trying to bootstrap a nonexistent operating system can cause this error to occur. See Section 5 of this manual for a table of workstation numbers. Check the following:

- The existence of the *[Sys]<Sys>WSnnn>SysImage.sys* file at the server workstation for the workstation type selected (nnn). If *[Sys]<Sys>WSnnn>SysImage.sys* cannot be found, the default system image file *[Sys]<Sys>WS>SysImage.sys* is loaded.
- The cluster communications cables, connections, and terminations
- The I/O board (processor module)
- The cluster communication circuitry

**B4** Bad checksum of system image.

The system image transferred from the server workstation is not a valid run file. Either the file is invalid, or the transmission was faulty or incomplete. Check the following:

- Whether the operating system of the cluster workstation is invalid
- The communications cables, connections, and terminations
- The cluster communication circuitry
- The I/O board (processor module)
- The CPU board (processor module)
- The memory board (processor module)

**B5** Read error.

Excessive input/output errors occurred while the bootstrap interface block was being read. Check the following:

- The I/O board (processor module)
- The cluster communications cables, connections, and terminations
- The cluster communications circuitry and CPU board

**B6** Read timeout.

During a read operation, no response was received from the server workstation. Check the following:

- The communications cables, connections, and terminations
- The cluster communications circuitry
- The operating system of the server workstation, which may have failed

**B7** Write DMA count is bad.

After completion of a write operation, the boot ROM determined that the entire block was not sent. Check the following:

- The I/O board (processor module)
- The CPU board (processor module)

**B8** Write timeout.

A write operation did not properly complete. Check the following:

- The cluster communication circuitry
- The CPU board
- The I/O board (processor module)

**B9** Bad bootstrap block format.

A bootstrap block of an invalid length was received. Check the following:

- The CPU board
- The format of the bootstrap file
- The cluster communication circuitry

**BA** DMA error.

After initializing the DMA channel for a read or write operation, the DMA controller did not contain the same information that was written to it.

Check the CPU board (processor module) and DMA logic.

**BB** Register errors generated while testing the IVGA gate array.

Check the motherboard and the optional video memory expansion (if any).

**BC** Incorrect vertical or horizontal retrace interval.

Check the motherboard and the optional video memory expansion (if any).

**BD** Color plane logic error detected.

Check the motherboard.

**BE** Video memory test failure.

Check the following:

- The video memory board
- The SIMMs
- The motherboard

**BF** DAC register error

Check the motherboard and the optional video memory expansion (if any).



- C0** Incorrect video output levels.  
Check the following:
- The video memory board
  - The SIMMs
  - The motherboard
- C1** GHA function error detected.  
Check the motherboard and the optional video memory expansion (if any).
- C2**  
**to**  
**DF** Reserved.
- E0** ROM did not check sum.  
There is a bad ROM (on B27, 2732 ROM) on the motherboard. This error is displayed on the keyboard LEDs, not the screen. On a B28-EXP, the system reports this error only on the speaker.  
Check the motherboard (processor module) and the boot ROM.
- E1** RAM error.  
There is a failure in the boot ROM's RAM work area. The boot ROM uses this work area to compose error codes. If a failure occurs, the error appears on the keyboard LEDs, and may also appear on the screen. On a B28-EXP, the system reports this error only on the speaker.  
Check the CPU board (processor module) and the memory boards.  
SG2100-320: This indicates an error in the working memory used by the boot ROM (locations 0C00h to 1A00h). Check the single inline memory modules (SIMMs) and the motherboard.

**E2** RAM error reading and writing zeros.

Check the memory boards, expansion RAM, and the CPU board. See E4 for more information.

SG2100-320: This error indicates a memory failure while reading and writing zeros to RAM. The address of the failure, the expected value, and the actual value read are displayed on the monitor screen. One of the single inline memory modules (SIMMs) probably failed. Run diagnostics to isolate the failure.

**E3** Error reading and writing ones (1).

Check the memory boards and expansion RAM and the CPU board. See E4 for more information.

SG2100-320: This error indicates a memory failure while reading and writing ones to RAM. The address of the failure, the expected value, and the actual value read are displayed on the monitor screen. Check the single inline memory modules (SIMMs). Run diagnostics to isolate the failure.

**E4** RAM error reading and writing addresses.

This explanation applies to codes E2, E3, and E4. An error occurred during the read and write RAM test. All 1s, all 0s, or an address pattern are written, read, and compared. The comparison showed that the data written and read were not identical. If a failure is confined to a small number of memory locations, the likely source of the failure can be derived from the high-order byte of the failed addresses. Following are the high-order addresses in the processor module, with the exceptions as noted:

0 to 3	Memory board
4 to 7	First RAM expansion cartridge
8 to B	Second RAM expansion cartridge
C to F7	Third RAM expansion cartridge
F8 to F9	Video RAM
FA to FB	Font RAM

For a B27 workstation, the addresses are:

0 to 7            Memory board  
8F                RAM expansion cartridge

B39 processors have no video RAM or font RAM, and have 1 MB base memory.

The error display for E2, E3, E4, and E5 (below) is:

```
E:  E2
1000:675C  0000  0002
```

In this display the symbols have the following meanings:

E: E2            Error code  
1000:675C       Hexadecimal address  
0000            Expected value  
0002            Received value

Check the following:

- The memory boards and expansion RAM
- The CPU board (processor module)
- The motherboard for any module connected on the X-Bus (or the F-Bus on a B27)

For a B28-EXP, the system displays the following:

```
E:  E4            Error code
Sa: Ra           Wrote: eeee  Read: aaaa
```

where

*Sa: Ra*        Is the segment address and offset of the failed word.  
*eeee*           Is the expected value (value written).  
*aaaa*           Is the actual value (value read).

For an SG2100-320, this error indicates a memory failure while reading and writing ones to RAM. The address of the failure, the expected value, and the actual value read are displayed on the monitor screen. Check the motherboard and the single inline memory modules (SIMMs). Run diagnostics to isolate the failure.

**E5** Error verifying RAM addresses.

An error occurred during the RAM addressing test. After completion of the RAM read/write address test, each RAM word should contain the sum of its own address. The RAM address test verifies that this is still true after one complete cycle of the test.

This error can be caused by a short or a shorted address line allowing different RAM locations to respond to the same CPU. It may also be caused by memory that picks up or drops bits when idle. Check the following:

- The memory board (processor module)
- The RAM expansion cartridges (processor module)

For an SG2100-320, this error indicates a failure while verifying a write to a particular address. Check the motherboard and the single inline memory modules (SIMMs).

**E6** Keyboard initialization error. (Different for B27. See below.)

An error occurred while the bootstrap ROM was initializing the hardware. Check the following:

- The video board (processor module)
- The I/O board (processor module)

SG2100-320: This error indicates a failure of the 82C106 keyboard controller on the motherboard. Check the motherboard.

B27: RAM parity error.

A parity error occurred during a memory read.

Check the memory board and the memory expansion module.

**E7** Nonmaskable interrupt (NMI) failure. (Different for B27. See below.)

An NMI was expected during X-Bus poll, but was not received. Check the following:

- The X-Bus modules
- The motherboard and the CPU boards

B27: Bad parity controller.

The parity controller cannot detect parity errors and/or indicates errors when none exist.

Check the memory board.

**E8** An unexpected nonmaskable interrupt (NMI) was received. (Different for B27. See below.)

When E8 appears on the screen, three hexadecimal words are displayed, as shown below:

```
E8
AAA  BBCC  DDEE
```

where

**AAA** Is the lower 16 bits of the address latched during the NMI.

**BB** Is "don't care" for a B38 or B39.

**CC** Is the upper 8 bits of the addresss latched during the NMI.

**DD** Applies to the B38 and B39 only, as described below:

	Bit				Bit			
DD	15	14	13	12	11	10	9	8
A25	A24	DMA	PE	B3	B2	B1	B0	

where

B3 - B0 are the 80386 byte definition bits. Refer to the Intel 80386 documentation for more information

PE is 1 if the NMI is due to memory parity, 0 if not due to memory parity

DMA is 1 if DMA was active, 0 if DMA was not active

A25, A24 are the upper two bits of the memory address

*EE* Is "don't care" for the B39.

B27: Bad 8251 keyboard USART.

The 8251 keyboard USART is never ready to transmit.

Check the CPU board.

**E9** Utility register error.

There is a failure on the 286i, 386i, or B39 motherboard.

Specifically, the utility register ROM bit (bit 2) was set when it should not be. The utility register is port 802Eh on the 286i, 386i, and B39.

Bad 8259 interrupt controller.

There are no external interrupts generated.

Check the CPU board.

**EA** Video boot load error.

A failure occurred when loading video boot code from an EPROM on a video card. This error may be caused by a bad video. Check the following:

- The video card
- The CPU board

B27: Keyboard timeout.

Nothing was received from the keyboard during initialization.

Check the following:

- The I/O board
- The video board
- The keyboard
- The monitor



**EB** Master DMA chip failure.

The master DMA ASIC failed a read/write register test. This error applies to the B39 only.

B27: Bad video RAM.

A data error occurred during the video RAM test. Check the following:

- The video board
- The CPU board
- The motherboard

**EC** Subsidiary DMA chip failure.

The subsidiary DMA ASIC failed a read/write register test. This error applies to the B39 only.

B27: Keyboard loopback failure.

The data received in loopback mode is incorrect. Check the following:

- The I/O board
- The video board
- The keyboard
- The monitor

**ED** B27: Bad font RAM.

A data error occurred during the RAM test. Check the following:

- The I/O board
- The motherboard
- The video board

## Boot ROM Codes for X-Bus Workstations and the ClientCard

---

- EE**      **B27: No F-bus timeout.**
- There was no bus timeout, or no NMI was generated by the timeout.  
Check the following:
- The memory board
  - The I/O board
- EF**      **B27: Bad 8254 timer.**
- The 8254 timer is not counting.  
Check the CPU board.
- F0**      **B28-EXP: RAM Parity Error.**
- If the error occurs during the firmware RAM memory test, the error code is reported only on the speaker. Otherwise, several hexadecimal words are displayed, as shown below:
- ```
E: F0      Error code
Sa:Ra      STAT: ssss      ADDR: aaaa
```
- where
- Sa:Ra*      Is the segment address and offset where the NMI occurred.
- ssss*      Is the memory parity controller status register value.
- aaaa*      Is the memory parity controller address register value.
- F1**      **B28-EXP: Bad 8251.**
- F2**      **B28-EXP: Bad 8259 interrupt controller.**
- There are no external interrupts generated.  
Check the CPU board.
- F3**      **B28-EXP: Keyboard timeout.**
- The keyboard is inoperative. Check the keyboard connections.

- F4**      **B27: DMA data error.**
- The data transferred by the 8237 DMA controller is incorrect. Check the following:
- The motherboard
  - The memory board
  - The CPU board
- B28-EXP: Bad video RAM.**
- A data error occurred during the video RAM test. On a B28-EXP, the system reports this error only on the speaker. Check the following:
- The video board
  - The CPU board
  - The motherboard
- F5**      **B27: DMA status error.**
- The status of the 8237 DMA controller after transfer is incorrect. Check the following:
- The memory board
  - The CPU board
- B28-EXP: Keyboard loopback failure.**
- The data received in loopback mode is incorrect. Check the following:
- The I/O board
  - The video board
  - The keyboard
  - The monitor
- F6**      **B28-EXP: Font RAM.**
- The system reports this error only on the speaker.
- F7**      **B28-EXP: Bad 8254 timer.**
- The 8254 timer is not counting. Check the CPU board.

## Boot ROM Codes for X-Bus Workstations and the ClientCard

---

- F8** B28-EXP: Bus Timeout Test failure.  
There was no bus timeout, or no NMI was generated by the timeout.
- F9** B28-EXP: Invalid monitor ID.  
The system reports this error only on the speaker.
- FA** B27: Bad video horizontal retrace.  
There is no indication of horizontal retrace occurring. Check the following:
- The video board
  - The CPU board
- B28-EXP: UCV chip failure.  
The system reports this error only on the speaker.
- FB** B27: Bad video vertical retrace.  
There is no indication of vertical retrace occurring.  
Check the following:
- The video board
  - The CPU board
- B28-EXP/B28-EX1: X-Bus ROM load failure.  
B28-LCW: RS-485 diagnostic.  
Write timeout.  
A write operation did not properly complete. Check the following:
- The cluster communication circuitry
  - The CPU board
  - The I/O board (processor module)
- FC** B27: Bad video blanking.  
There is no indication of blanking occurring. Check the following:
- The video board
  - The CPU board

**B28-EXP/B28-EX1:** X-Bus memory map and CPU memory conflict.

Remove the CPU memory expansion cards that bring the RAM above 8MB.

**B28-LCW:** RS-485 diagnostic.

Transmit timeout.

EISA/ISA-bus workstations with ClientCard: Error reading from the floppy or hard disk. Check the disk to see that it is mounted.

**FD** **B27:** Bad video pixel.

The pixel pattern is incorrect. Check the following:

- The video board
- The CPU board

**B28-LCW:** RS-485 diagnostic.

Bad DMA write count.

EISA/ISA bus workstations with ClientCard: Error in the floppy or hard disk boot sector. Could not boot. Try another disk.

**FE** **B27:** Bad real time clock.

The real time clock is not counting. Check the following:

**B28-LCW:** RS-485 diagnostic.

Receiver error.

EISA/ISA-bus workstations with ClientCard: Switches on ClientCard set to invalid I/O address. Call Technical Support. The ClientCard switches need to be reset.

**FF** **B27:** No real time clock interrupt.

There are no interrupts generated by the real time clock. Check the CPU board.

### Boot ROM Codes for B24 Workstations

During the hardware tests and bootstrap for the B24 workstation, the motherboard LEDs display a code that indicates which test is being performed. The codes are as follows, where 1 indicates On and 0 indicates Off.

| <b>LEDs</b> | <b>Tests Being Performed</b>              |
|-------------|-------------------------------------------|
| 1 1 1 1     | Start of microprocessor initialization    |
| 1 1 1 0     | Start of firmware memory test             |
| 1 1 0 1     | Start of ROM checksum test                |
| 1 1 0 0     | Start of video memory and controller test |
| 1 0 1 1     | Start of MM 58274 RTC test                |
| 1 0 1 0     | Start of 8259A interrupt controller test  |
| 1 0 0 1     | Start of 2681 USART I-Bus/S-Bus test      |
| 1 0 0 0     | Start of Keyboard/I-Bus test              |
| 0 1 1 1     | Start of NVM test                         |
| 0 1 1 0     | Start of 8254 timer test                  |
| 0 1 0 1     | Start of RS-232-C (A & B) 7201 USART test |
| 0 1 0 0     | Start of RS-232-C (C & D) 2681 USART test |
| 0 0 1 1     | Start of low RAM test                     |
| 0 0 1 0     | Start of high RAM test                    |
| 0 0 0 1     | Start of security module test             |
| 0 0 0 0     | Menu mode and boot                        |



In addition to the above 4-bit LED codes displayed on the motherboard, once the screen has been tested and found to be working, an 8-bit error code is displayed for each error found. As for the other workstations, these 8-bit codes are displayed as two hexadecimal digits. The codes and their interpretations are listed below:

**00**

to

**A1**

Reserved.

**A2**

Communications initialization error.

The workstation was never polled. No frames are being sent by the server.

Check:

- The motherboard
- The server workstation
- The communications cable
- The server operating system, which may have crashed

**A3**

SIO communications error.

The communications line is always busy. The workstation cannot send without causing a collision.

Check:

- The motherboard
- The server workstation
- The communications cable
- The server operating system, which may have crashed

**A4**

to

**A8**

Reserved.

## Boot ROM Codes for B24 Workstations

---

**A9** No identification number available. The initialization routine monitored the cluster communications line, but was not able to find a free identification number. This is usually caused by attaching more workstations to a cluster communications line than the operating system of the server is designed to accept.

Check the server workstation operating system, which may have crashed.

**AA** Identification failure.

The initialization routine found free workstation identification numbers by monitoring the communications line, but errors were detected when it tried to use one. This is usually caused by a failure of the collision retry algorithm and can be overcome by resetting each machine involved in the collisions.

**A2** Communications initialization error.

The server was never polled. No frames are being sent by the server. Check the following:

- The motherboard
- The server
- The communications cable
- The server operating system, which may have crashed

**A3** SIO communications error.

The communications line is always busy. The workstation cannot send without causing a collision: Check the following:

- The motherboard
- The server workstation
- The communications cable
- The server operating system, which may have crashed

**A4**

to

**A8** Reserved.

- A9** No identification number available.
- The initialization routine monitored the cluster line, but was not able to find a free identification number. This is usually caused by attaching more workstations to a cluster line than the operating system of the server is designed to accept.
- Check the server operating system.
- A9** Identification failure.
- The initialization routine found free workstation identification numbers by monitoring the cluster line, but errors were detected when it tried to use one. This is usually caused by a failure of the collision retry algorithm.
- Reset each workstation involved in the collision.
- AB to AF** Reserved.
- B0** No UP frame.
- After transmitting a frame, the server did not respond with a UP. A different frame type, or a bad frame was detected. Check the following:
- Whether a cluster workstation is using the fixed identification mode
  - The motherboard
- B1** Read UI error.
- A bootstrap block (frame type UI) was expected, but another type was received. Check the motherboard.
- B2** Reserved.

## Boot ROM Codes for B24 Workstations

---

### **B3** Disconnect during boot.

The server workstation chose to send a DISC frame because of a conflict with the system image file or because of extensive errors during transmission. Trying to boot a nonexistent operating system can cause this error. Operating system number 200 is used for the B24. Check the following:

- *[Sys]<Sys> Ws200>SysImage.sys* file at the server workstation
- The communication cable
- The motherboard

### **B4** Bad checksum of system image.

The system image transferred from the server workstation is not a valid run file. Either the file is invalid, or the transmission was faulty or incomplete. Check the following:

- The operating system of the cluster workstation to see if it is valid.
- The motherboard.
- The communication cable.

### **B5** Read error.

Excessive I/O errors occurred while the bootstrap interface block was being read. Check the following:

- The motherboard
- The communication cable

### **B6** Read time out.

During a read operation, no response was received from the server workstation.

Check the server workstation operating system, which may have crashed.

### **B7** DMA write count bad.

After completion of a write operation, the boot ROM determined that the block was not sent entirely.

Check the motherboard.

- B8** Write time out.  
A write operation did not complete properly.  
Check the motherboard.
- B9** Bad boot block format.  
A bootstrap block of invalid length was received.  
Check the format of the bootstrap file.

### 2681 Error Codes

The error codes C0 to CB come in pairs. The first listed code is for the 2681 controlling the I and S buses. The second listed code is for the 2681 controlling RS-232 ports C and D.

- C0**  
to  
**C1** TxRDY error.  
During the test of the 2681, the TxRDY bit did not reflect the expected status.  
Check the motherboard.
- C2**  
to  
**C3** TxEMT error.  
During the test of the 2681, the TxEMT bit did not activate when expected.  
Check the motherboard.
- C4**  
to  
**C5** RxRDY error.  
During the test of the 2681, the RxRDY bit did not reflect the expected status.  
Check the motherboard.

## Boot ROM Codes for B24 Workstations

---

- C6**  
to  
**C7**      **Bad character.**
- During the test of the 2681, a bad character was received during the internal loopback.
- Check the motherboard.
- C8**  
to  
**C9**      **RxOver set.**
- During the test of the 2681, the RxOver bit did not reflect the expected status.
- Check the motherboard.
- CA**  
to  
**CB**      **No interrupt generated.**
- During the test of the 2681, no TxEMT interrupt was generated.
- Check the motherboard.

### Keyboard Test Error Codes

- CC**      **Keyboard did not answer.**
- During the test of the keyboard, no answer was received. Check the following:
- The motherboard
  - The keyboard
  - The monitor
- CD**      **Bad echo.**
- During the echo test of the keyboard, a character was sent and a different character was received. Check the following:
- The motherboard
  - The keyboard
  - The monitor



**CE**      Bad ROM checksum.

During the keyboard checksum test, an incorrect answer was returned. This can be caused by pressing a key during the test. Check the following:

- The motherboard
- The keyboard
- The monitor

**CF**      Write timeout.

During the keyboard test, an error occurred when writing.

Check the following:

- The motherboard
- The keyboard
- The monitor

**D0**      No keyboard found.

During initialization, no keyboard was found connected to the system. This does not necessarily reflect an error. Check the following:

- The motherboard
- The keyboard
- The monitor

### ROM Checksum Test Error Codes

**D1**      ROM checksum error.

A bad checksum was calculated for the CPU ROMs. Check the following:

- The motherboard
- The ROM chips

### RTC Test Error Codes

**D2**      Bad status.

During the test of the RTC, the status flags (interrupt occurred and data changed) did not accurately reflect the status of the RTC. The flags were always detected set. Check the following:

- The motherboard
- The battery supplying the RTC

**D3**      Bad status.

During the test of the RTC, the status flags (interrupt occurred and data changed) did not accurately reflect the status of the RTC. The flags were always detected reset. Check the following:

- The motherboard
- The battery supplying the RTC

**D4**      Bad value.

During the test of the RTC, no change was detected in the one-tenth of a second register value. Check the following:

- The motherboard
- The battery supplying the RTC

### Interrupt Test Error Codes

**D5**      No interrupt.

During the interrupt test, the 8259 interrupt controller did not generate an interrupt.

Check the motherboard.

### VSDD Error Codes

**D6**      Bad ATAC.

During the VSDD test, the ATAC did not change values. Check the following:

- The motherboard
- The monitor

The listed error code gives an indication of which timer was found to be faulty. The possible errors are bad status, bad counter values, or bad output signals.

**D7**      Timer 0 error.

During the timer test, timer 0 of the 8254 did not pass one of the three mentioned tests.

Check the motherboard.

**D8**      Timer 1 error.

During the timer test, timer 1 of the 8254 did not pass one of the three mentioned tests.

Check the motherboard.

**D9**      Timer 2 error.

During the timer test, timer 2 of the 8254 did not pass one of the three mentioned tests.

Check the motherboard.

### 7201 MPSC Error Codes

**DA**      TxEMT error.

The TxEMT bit did not accurately reflect the state of the chip, during the test of the 7201 serial Input/Output (SIO) chip.

*Note:*    This error can be caused by a nonworking 8254, since 8254 is used to provide a clock for the 7201.

Check the motherboard.

## Boot ROM Codes for B24 Workstations

---

**DB** No interrupt.

During the test of the 7201 SIO, no TxEMT interrupt was generated.

*Note: This error can be caused by a nonworking 8254, since the 8254 is used to provide a clock for the 7201.*

Check the motherboard.

### Optional 2681

**DC** Optional 2681 does not exist.

During initialization, the optional 2681 was not found. This is not necessarily an error.

### Communications Error Codes

**DD** Channel A loopback error.

During the external loopback test of the 7201, channel A did not receive the same character which was sent. Check the following:

- The motherboard
- The communication loopback plugs

The external loopback plugs are necessary for successful completion of this test.

**DE** Channel B loopback error.

During the external loopback test of the 7201, channel B did not receive the same character that was sent. Check the following:

- The motherboard
- The communications loopback plugs

The external loopback plugs are necessary for successful completion of this test.

**DF** Channel C loopback error.

During the external loopback test of the 2681, channel C did not receive the same character which was sent. Check the following:

- The motherboard
- The communications loopback plug

The external loopback plugs are necessary for successful completion of this test.

**E0** Channel D loopback error.

During the external loopback test of the 2681, channel D did not receive the same character that was sent. Check the following:

- The motherboard
- The communication loopback plugs

The external loopback plugs are necessary for successful completion of this test.

**E1** Cluster communications loopback error.

During the loopback test of the cluster communications channel, the received frame did not match that which was sent. Check the following:

- The motherboard
- The cluster communications line

The B24 must be disconnected from the cluster line for successful completion of this test.

### Security Module Test Error Codes

**E2** Security module not present. During initialization, no security module was detected. This is not necessarily an error. Check the following:

- The motherboard
- The security module

- E3** Security module did not respond.
- During initialization a security module was detected, but communication with it was not possible.
- Check the following:
- The motherboard
  - The security module
- E4** Security module CPU error.
- During initialization, the security module's CTR reported a problem with its CPU or ROM. Check the following:
- The motherboard
  - The security module
- E5** Security module RAM error.
- During initialization, the security module's CTR reported a problem with its RAM. Check the following:
- The motherboard
  - The security module
- E6** Security module data cipher processor (DCP) error.
- During initialization, the security module's CTR reported a problem with its DCP. Check the following:
- The motherboard
  - The security module
- E7** Security module FIFO error.
- During the test of the security module, the FIFO buffer did not function correctly. Check the following:
- The motherboard
  - The security module



**E8** Security module encryption error.

During the test of the security module, a text was encrypted incorrectly. Check the following:

- The motherboard
- The security module

**E9** S-Bus loopback error.

During the loopback test of the S-Bus, the character that was received did not match the character that was sent. Check the following:

- The motherboard.
- The S-Bus loopback plug.

The S-Bus loopback plug is necessary for successful completion of this test.

### **NVM Test Error Codes**

**EA** APF line is always active.

During the test of the CMOS RAM, the advanced power fail line was always active. Check the following:

- The motherboard
- The battery power for the CMOS RAM

### **RAM Test Error Code**

**EB** Bad address line.

During a memory test, an address line was found to be inoperative. Check the motherboard.

**EC** Could not verify all 1s.

See error code EF for explanation.

**ED** Could not verify word address.

See error code EF for explanation.

## Boot ROM Codes for B24 Workstations

---

**EE** Could not verify all 0s.

See error code **EF** for explanation.

**EF** Bad memory location.

This explanation applies to errors **EC**, **ED**, **EE**, and **EF**. During the memory test, the verification of read and write data was not successful. The comparison showed that the data written and read back were not identical. The location of the bad RAM may be determined from the segment value of the error. The following are the various RAM locations:

|              |               |
|--------------|---------------|
| 0000 to 7FFF | Basic RAM     |
| 8000 to EBFF | Expansion RAM |
| EC00 to EFFF | CMOS RAM      |
| F000 to F7FF | Video RAM     |

The following display is for these memory errors:

Failure: VV

WWWW: XXXX YYY YZZZ

where

VV Is an error code.

WWWW Is the segment value of the error.

XXXX Is the offset of the error.

YYYY Is data written to the memory location.

ZZZZ Is data read back.

Check the motherboard.

**F0** Parity error during memory test.

An unexpected NMI was generated as a result of reading from or writing to memory.

Check the motherboard.

### Boot Errors for EISA and ISA Workstations

EISA/ISA workstations, such as the SuperGen Series 3000, boot differently than other CTOS workstations, as described at the beginning of this section. EISA/ISA workstations contain a BIOS that reports errors to the screen or as a series of beep codes. See the hardware installation manual for your workstation for details on BIOS errors.

Some CTOS initialization errors, especially when booting across the cluster, can be reported from the ClientCard. These are listed in the earlier section entitled, "Boot ROM Codes for X-Bus Workstations and the ClientCard."

You can also see errors from the system Boot Record when an EISA or ISA workstation is booted. These errors are listed below:

- E1**      Checksum error in *[Sys]<Sys>Sysimage.sys*.  
An error was encountered reading a *[Sys]<Sys>Sysimage.sys*. You may have a corrupt system image file.
- E2**      Int 13 error reading *[Sys]<Sys>Sysimage.sys*.  
This error indicates a problem with hardware or hardware configuration. Run Setup or the System Configuration Utility to check your hardware configuration.
- E3**      Volume Home Block not found.  
The disk you are trying to boot from has no Volume Home Block. It may not have been initialized or it may have been corrupted. Try initializing the disk with the **Format Disk** command.
- E4**      Signature not found in run file header.  
The proper identification was not found in the run file header. You may have a non-system disk in the floppy disk drive, or the hard disk drive you are booting from may not contain a bootable system image.
- E5**      *[Sys]<Sys>Sysimage.sys* is too big.  
The *[Sys]<Sys>Sysimage.sys* file found could not be loaded because of lack of memory.

### Boot ROM Codes for SuperGen Series 5000 Workstations

The Series 5000 boot ROM performs a self-test when the workstation is powered up. This is a set of diagnostic tests that reside in the ROM-based program that handles low level system operations and acts as an interface between operating systems and the hardware.

The SuperGen Series 5000 workstations include a seven-segment LED. The boot ROM error codes appear on the seven-segment display in the following sequence:

- First digit of the error code
- Second digit of the error code
- - (dash)
- First digit of the error code
- Second digit of the error code

This pattern is repeated four times for nonfatal errors and then the confidence test continues. For fatal errors, the error code cycles through the display sequence.

After the video has been initialized, boot ROM errors are sent to the video display. Error codes are displayed on both the monitor screen and on the seven-segment display when a BIOS diagnostic test fails. Note that for some fatal errors, the system can hang and the error code cannot be displayed.

When the error code represents a memory error, the error code is displayed on the seven segment display. In addition, the physical address, the expected value, and the actual value read are displayed on the screen. The format for memory errors displayed on the screen is as follows:

```
E:NN Address:Offset      Expected Value:Value Read
```

Error conditions may be displayed on the screen, the seven-segment LEDs, or both. The screen errors are displayed in the following manner:

E: XX/YY

where

**XX** Is the compatible CTOS status code number, if one exists.

**YY** Is the BIOS status code specific to the Series 5000.

**Note:** *Only the BIOS status codes are listed in this subsection.*

More than one error code may be displayed during the self test. Pay attention to all of the errors displayed. Observing the test sequence may help to isolate a failure.

Some status codes list more than one possible cause. The most probable cause is listed first.

The status codes are given in hexadecimal format.

- 00** Reserved.
- 01** Processor register test in progress.  
A data pattern is cycled through all registers of the processor and verified to ensure that the processor can be addressed.  
Check the CPU logic cartridge.
- 02** CMOS write/read test in progress.  
CMOS RAM on the CPU board is verified.  
Check the CPU logic cartridge.
- 03** BIOS ROM checksum in progress.  
The checksum of the BIOS ROM is verified.  
Check the CPU logic cartridge.
- 04** Programmable Interval Timer (PIT) initialization in progress-timer 1.  
Timers are initialized for correct CPU timed events.  
Check the motherboard.

## Boot ROM Codes for SuperGen Series 5000 Workstations

---

- 05** DMA initialization in progress.  
DMA registers are initialized for data transfers.  
Check the motherboard.
- 06** DMA page register write/read test in progress.  
DMA registers are initialized for data transfers.  
Check the CPU logic cartridge.
- 08** RAM refresh verification in progress.  
The RAM refresh counter is initialized at the beginning of BIOS code to make sure data in memory is not lost when the system is reset.  
Note: This is an important feature used in a crash dump.  
Check the motherboard.
- 09** 1st 64K RAM test in progress.  
BIOS ROM initialization code is placed in memory. This code is used in subsequent POST routines.  
Check the memory logic cartridge.
- 0A** 1st 64K RAM chip or data line test in progress.  
BIOS ROM initialization code is placed in memory. This code is used in subsequent POST routines.  
Check the memory logic cartridge.
- 0B** 1st 64K RAM parity test in progress.  
BIOS ROM initialization code is placed in memory. This code is used in subsequent POST routines.  
Check the memory logic cartridge.
- 0C**  
**to**  
**0D** Reserved.



- 0E**      Fail-safe timer test failed.
- In this test, the fail-safe timer in the EISA chip set on the motherboard is loaded with a preset value. The BIOS then waits for a nonmaskable interrupt (NMI) to occur on the counter's timeout. If a fail-safe timer NMI occurs, then the test is successful. If a fail-safe timer NMI does not occur, the test has failed, and this error is reported. This test routine assumes that the first 64K RAM test is successful and the BIOS can use working memory.
- Check the motherboard.
- 0F**      Software NMI port test failed.
- The BIOS attempts an I/O operation on the software NMI port to generate a nonmaskable interrupt. This error is reported if an NMI does not occur.
- Check the motherboard.
- 10**  
**to**  
**1F**      Reserved.
- 20**      Subsidiary DMA register test failed.
- This test initializes the registers of the subsidiary DMA chip on the motherboard for data transfers.
- Check the motherboard.
- 21**      Master DMA register test failed.
- This test initializes the registers of the master DMA chip on the motherboard for data transfers.
- Check the motherboard.
- 22**      Master interrupt mask register test failed.
- This test writes data to the master interrupt mask register and reads the data to confirm that the test was successful.
- Check the motherboard.

## Boot ROM Codes for SuperGen Series 5000 Workstations

---

- 23**      Subsidiary interrupt mask register test failed.
- This test writes data to the master interrupt mask register and reads the data to confirm that the test was successful.
- Check the motherboard.
- 24**  
**to**  
**26**      Reserved.
- 27**      Keyboard controller test in progress.
- The keyboard controller on the video board is initialized. If the initialization fails, this error is reported.
- Check the video board.
- 28**  
**to**  
**33**      Reserved.
- 34**      Timer tick interrupt test failed.
- The timer should generate approximately 18 ticks per second. If the timer is not running, or is running at the wrong speed, this error is reported. This is not a fatal error, but will cause problems with functions related to the timer.
- Check the CPU board.
- 35**      Shutdown test failed.
- This test issues a shutdown command to allow the processor to enter real mode after running in protected mode. This routine preserves compatibility with 80286 based operating systems and is available to the operating system after the self-test is complete.
- Check the CPU board.
- 36**      Gate A20 failure.
- This test toggles gate A20 on the processor to ensure that the processor can switch from real mode to protected mode.
- Check the CPU board.

- 37** Unexpected interrupt occurred in protected mode.  
Check the CPU board.
- 38** RAM test failure above address 0FFFFh.  
This test writes a data pattern to extended memory (above 1MB).  
The data is then read back to verify the write.  
Check the CPU board.
- 39** Reserved.
- 3A** Programmable Interval Timer failure: timer 2.  
This error indicates a failure or programmable interval timer 2 on  
the motherboard.  
Check the motherboard.
- 3C** Serial port test failed.  
This test writes a value to a register in the serial controller and then  
reads the contents of the register to verify the write operation.  
Check the motherboard.
- 3D** Parallel port test failed.  
This test writes a value to a register in the logic controller for the  
parallel port and then reads the contents of the register to verify the  
write.  
Check the motherboard.
- 3E** Math coprocessor test failed.  
This test checks for the presence of a numeric coprocessor. If a  
numeric coprocessor is present, the coprocessor is tested. If the test  
fails, this error is reported.  
Check the CPU board.
- 3F** Reserved.

- 40** Time of day is not set or not valid.
- The real-time clock read the time of day data in CMOS RAM and checked the data to see if it is within valid boundaries. If the time of day data is out of bounds, CMOS data is corrupt, and this error is reported.
- Data in CMOS RAM may be corrupt; run Setup.
- 41** Time of day clock stopped.
- This test confirms that the time of day data in CMOS RAM is being updated by checking the interrupt flag register in CMOS RAM. If the interrupt flag is not being set by the real-time clock, this error is reported.
- Check the CPU board.
- 42** Configuration information mismatch, default configuration used.
- The configuration file in CMOS RAM does not match the configuration of your hardware. Default configuration data is used automatically. Run Setup and enter valid configuration information that matches your hardware configuration. You must then exit the panel menu and reboot (press the reset button) to use the new configuration information.
- This error recurs after a reset if the configuration information in the Setup menu does not match the hardware configuration.
- This error also occurs when you first install your base unit, when you remove and reinstall the CPU board, and may also occur when you change the battery.
- If your hardware configuration matches the default configuration, you see this error code once after reinstalling the CPU board or the battery
- If this error occurs repeatedly after cycling power on the base unit and the configuration information in Setup matches your configuration, the battery may be discharged.
- Check the following:
- Configuration mismatch (run Setup)
  - Discharged battery

- 43**      **Bad checksum of ROM.**  
This test runs a checksum of the ROM and stores the checksum value in ROM.  
Check the boot ROM (on the CPU board).
- 44**      **Invalid CMOS floppy configuration information.**  
The configuration information in CMOS RAM does not match the floppy drive type.  
Check for a configuration mismatch (run Setup).
- 45**      **CMOS battery dead.**  
This test reads the power-sense pin on the real-time clock (RTC) on the CPU board to determine whether the battery is connected to the real time clock. The battery supplies power to the real time clock and CMOS RAM when the workstation is turned off.  
Check for a discharged or disconnected battery.
- 46**  
**to**  
**4F**      **Reserved.**
- 50**      **Floppy disk subsystem reset failed.**  
This test resets the floppy drive controller in the floppy drive. This error indicates a failure of the floppy drive controller.  
If possible, run diagnostics on the floppy drive to confirm the failure and confirm that the floppy drive is properly installed.
- 51**      **Diskette drive 0 seek failure.**  
This test checks floppy drive 0 by performing a seek. This error indicates mechanical problems with the floppy drive module.  
Run diagnostics on the floppy drive to confirm the failure. This test is performed only if the floppy drive contains a floppy disk during the self-test.  
Check the floppy drive.

- 52** Floppy drive 1 seek failure.
- This test checks floppy drive 1 by performing a seek. This error indicates mechanical problems with the floppy drive module.
- Run diagnostics on the floppy drive to confirm the failure. This test is performed only if the floppy drive 1 contains a floppy disk during the self-test.
- Check the floppy drive.
- 53** Floppy disk read failure.
- This test performs a read of the floppy disk. This error could be caused by an incorrectly formatted or defective disk.
- If the error occurs repeatedly on a known good disk, suspect the floppy drive. Run diagnostics on the floppy drive to confirm the failure.
- 54** Not a bootable disk.
- You attempted to boot from a floppy disk that does not contain a bootable system image file.
- Make sure that the floppy disk has a bootable system image.
- 55** No boot device available.
- You attempted to boot from a floppy disk that is not formatted.
- Make sure that the floppy disk has been formatted.
- 56**  
**to**  
**5F** Reserved.
- 60** The partition table on the start-up drive is incorrect.
- This error is displayed when the unit attempts to boot from the hard disk and the master boot record on the hard disk finds inconsistency in the partition table located in the first sector of the hard disk. This error can occur if there is no partition table, no active partition, or if there is more than one active partition on the hard disk.
- Check the hard disk for a formatting problem or hard disk drive failure.



- 61** The operating system cannot be loaded from the start-up drive.  
This error is reported if the master boot record on the hard disk is unsuccessful when loading the boot sector of the active partition.  
Check the hard disk for a formatting problem or hard disk drive failure.
- 62** The operating system is missing from the start-up drive.  
This error is reported if the boot sector of the active partition is invalid (does not contain a signature). This error indicates that there is corrupt data in the boot sector of the hard disk.  
Check the hard disk for a formatting problem or hard disk drive failure.
- 63**  
**to**  
**6F** Reserved.
- 70** Memory parity error above first 64K of memory.  
A byte read from memory has an incorrect parity check value.  
If your workstation contains optional memory expansion beyond the base memory, remove the memory expansion and try to reproduce the error to isolate the failure. Check the following:
- Memory board
  - Optional memory expansion
- 71** Data line error.  
One or more data lines is not accessing memory.  
Check the memory board.
- 72** Odd/even logic error above first 64K of memory.  
This test writes to every other bit in a block of memory and then reads the contents of that block. The pattern is then shifted one bit and the test is run again to fully test the memory block.  
Check the memory board.

## Boot ROM Codes for SuperGen Series 5000 Workstations

---

- 73** Double word logic error above first 64K of memory.  
This test addresses two words in memory and then writes a data pattern to those two words. The data is then read back. This test confirms that 32-bit addressing is functional.  
Check the memory board.
- 74** High memory error.  
This test checks memory above 1MB.  
Check the memory board.
- 75** Address line error.  
This test checks memory other than system memory installed in the base unit, such as memory on the video board or an optional logic cartridge that has its own memory. If video memory is corrupt, the error code may appear on the LED, rather than on the screen. Also, when video memory is corrupt, the monitor may display garbage-like data.  
Remove suspect cartridges and reboot to see if the error recurs.
- 76** Memory write/read error above first 64K of memory.  
This test writes a data pattern to memory above 64K and reads the data back to verify the write.  
Check the memory board.
- 77**  
**to**  
**7F** Reserved.
- 80** Keyboard controller error.  
This test resets the keyboard controller on the video board. The error is reported if the reset fails.  
Check the video board.

- 81**      Keyboard null error.
- This test checks to see if a device is plugged into the keyboard port. This error is reported if the keyboard is not plugged in.
- Check the following:
- That the keyboard is plugged in
  - The video board
- 82**      Keyboard clock line error.
- This test performs a write to the keyboard controller clock registers. The error is reported if the write is not confirmed when read.
- Check the video board.
- 83**      Keyboard data line error.
- This test performs a write and a read of the keyboard controller registers in the keyboard.
- Check the video board.
- 84**      Stuck key keyboard error.
- This test reads the contents of the keyboard buffer several times. If the same character is read repeatedly, the error is displayed.
- Check the keyboard.
- 85**      Keyboard is locked.
- This test checks the contents of the keyboard control register. This error is displayed if the keyboard is locked.
- Check the keyboard.
- 86**  
**to**  
**8F**      Reserved.
- 90**      Shadow memory error encountered at:
- This memory test copies the BIOS code from ROM into system memory and then verifies the write. The address, expected value, and actual value read are reported to the screen. Check the memory board.

## Boot ROM Codes for SuperGen Series 5000 Workstations

---

- 91** ROM BIOS shadow memory failure.  
This error is displayed if the test fails while writing the BIOS code to shadow memory.  
Check the memory board.
- 92** Video BIOS shadow failure.  
This test writes the video ROM BIOS code from the video board to system memory. This error is displayed if the write fails.  
Check the memory board.
- 93** SRAM decoder initialization failure.  
The test checks the SRAM decoder on the memory board.  
Check the memory board.
- 94** DRAM at ROM address range failure.  
This test initializes memory on the memory board. This error is reported if initialization fails.  
Check the memory board.
- 95** SRAM test failure.  
Initialization of memory on the memory board failed.  
Check the memory board.
- 96** Internal cache test failure.  
This test checks the internal cache controller. This error is reported if the test fails.  
Check the CPU board
- 97**  
**to**  
**9F** Reserved.
- A0** Unresolved memory parity error.  
This test writes data to memory and reads the data checking the parity bits. If a parity bit is not set, the error is reported.  
Check the memory board.

- A1** Memory parity error at:  
This test writes data to memory, reads the data, and checks the parity bits. If a parity bit is not set, the error is reported. The address of the error is also displayed.  
Check the memory board.
- A2** I/O expansion board NMI error, slot #:  
This error indicates that a board installed in one of the expansion slots is generating a nonmaskable interrupt (NMI). The slot number is displayed.  
This error could be caused by an ISA or EISA board, or by another optional logic cartridge.  
Check in the slot indicated for a failed board.
- A3** I/O expansion board NMI error.  
This error indicates that an optional cartridge installed in one of the expansion slots is generating a nonmaskable interrupt.  
This error could be caused by an ISA or EISA board, or by another optional logic cartridge.  
Check the boards in expansion slots for a failure.
- A4** Expansion board was disabled.  
A board in one of the expansion slots was generating an unexpected interrupt during the I/O check performed by the BIOS.  
Check the ISA or EISA board.
- A5** Fail safe timer NMI error.  
This error is reported when a system nonmaskable interrupt occurs. This is a fatal interrupt; the system will fail.  
Check the CPU board.
- A6** Unresolved bus timeout NMI error.  
This error represents a nonmaskable interrupt generated on the ISA or EISA bus.  
This error is caused by the ISA or EISA board that is using the bus at the time. Remove suspect boards and reboot to isolate the failure.

## Boot ROM Codes for SuperGen Series 5000 Workstations

---

- A7** Bus timeout NMI error, slot #:  
This error represents a nonmaskable interrupt generated on the ISA or EISA bus. The slot number is indicated on the display.  
Check the indicated ISA or EISA board.
- A8** Software NMI error.  
This error indicates a software generated nonmaskable interrupt. No hardware failure is indicated by this error.  
Check for a software error.
- A9** Invalid configuration storage error.  
The BIOS writes a data pattern to CMOS RAM and reads the data back to verify the write. This error is displayed if the data in CMOS RAM does not compare.  
Check the CPU board.
- AA** Configuration error for slot #:  
This error is displayed if the configuration tables stored in CMOS RAM for EISA or ISA boards don't match the actual hardware configuration. The slot number for the mismatch is given. This error can show up if you move a board from one expansion slot to another.  
Check for a configuration violation.
- AB** ID information mismatch for slot #:  
This error is displayed if the ID information stored in CMOS RAM for EISA or ISA boards doesn't match the actual hardware configuration. The slot number for the mismatch is given. This error can show up if you move a board from one expansion slot to another.  
Check for a configuration violation.
- AC**  
**to**  
**AF** Reserved.



- B0** Video display adapter failed; using alternate, CR, LF, 0
- This test checks for the presence of a display adapter. The error indicates the failure of the display adapter. Confirm that the video cartridge is correctly installed.
- Check the video board.
- B1** Screen does not blank.
- B2**
- to**
- BF** Reserved.
- C0** Unexpected software interrupt: #
- This test reads the software interrupt register. If an unexpected interrupt or an illegal software generated interrupt occurs, the interrupt number is reported to the screen. This error is reported while the operating system is running; it is not a boot error.
- Check for a software error.
- C1** Unexpected hardware interrupt: #
- This test reads the hardware interrupt register. If there is an unexpected interrupt, the error is reported. The interrupt number is reported to the screen. Check for a hardware configuration violation.
- C2** I/O card parity error.
- This error indicates that an optional I/O card, such as an E-Net board, is failing a parity check on the I/O data.
- Check the I/O board.
- C3** Memory parity error.
- This error indicates a failed memory parity check on an optional I/O board, such as an E-Net board, installed in the base unit.
- Check the I/O board.

- C4** Cannot crash dump onto device.  
This error is reported if a memory dump is attempted and the file *<Sys>CrashDump.sys* does not exist.  
This is a software error; no hardware failure is indicated by this error.
- C5** Device write error during crash dump.  
This error is reported if a crash dump is attempted and the target volume does not exist.  
Make sure that the target volume is mounted.
- C6**  
to  
**CF** Reserved.
- D0** Auxiliary device failure error 8602.  
This error indicates a failure of the mouse or indicates that the wrong device is plugged into the mouse port on the back of the SuperGen 5000. Check the following:
- The mouse
  - The mouse port, for an improper device
- D1**  
to  
**DF** Reserved.
- E0** External timer failed.  
This error indicates a failure of one of the external timers on the CPU board. This failure does not indicate a failure of the processor.  
Check the CPU board.
- E1** External DMA failed.  
This error indicates a failure of an external DMA controller on the CPU board. (This error does not indicate a failure of the timers on the processor itself.)  
Check the CPU board.

**E2** External interrupt controller failed.

This error indicates a failure of the external interrupt controller on the CPU board. This failure does not indicate a failure of the processor itself.

Check the CPU board.

**E3**

to

**FF**

Reserved.

### Boot ROM Codes for SG2100-320 Workstations

SG2100-320 workstations report many of the same boot errors as X-Bus workstations. See “Boot ROM Codes for X-Bus Workstations and ClientCards,” earlier in this section. In addition the following sequence is displayed on the seven segment display during boot ROM initialization:

- 1           The 8254 timer is programmed for memory refresh.  
  
              The memory controller is programmed and memory registers are set. If the unit hangs on this step of the sequence, check the motherboard.
- 2           The CPU registers are read for saved values.  
  
              Workstation hardware is identified and initialized. If the unit hangs on this step of the sequence, check the motherboard.
- 3           The boot ROM checksum test is run.  
  
              If the unit hangs on this step of the sequence, it indicates that the ROM may be bad. Check the motherboard.
- 4           Working memory is tested for use by the boot ROM.  
  
              The boot ROM uses memory from address 0C00 to 1A00. If the unit hangs on this test, check the single inline memory modules (SIMMs) or the motherboard.
- 5           The video RAM and video controllers are tested and video is initialized.  
  
              If the unit hangs on this test, check the video memory expansion (if any), or the motherboard. Try to duplicate the failure with the video memory expansion removed if you suspect that it has failed.
- 6           The video registers are set with final values.  
  
              If the unit hangs on this step of the initialization sequence, check the video memory expansion (if any), or the motherboard.

- 7** The keyboard is reset and initialized.
- The boot ROM determines whether the **SPACEBAR** is depressed. If so, the panel menu is displayed. If the user then types the character **B** (while the panel menu is displayed), the memory test is bypassed.
- If the **SPACEBAR** is not depressed, memory is tested. The display shows the following while the memory test runs.
- T**  
\*\*\*\*\*
- 8** The boot ROM system initialization is complete. Any errors detected are reported at this point.





# Appendix B

## Shared Resource Processor Front Panel and Processor Status Codes

### Front Panel Status Display

The status of system startup operations is reported through the front panel status display on the base enclosure of the shared resource processor. The status display shows a series of two-character codes indicating the progress of the boot-procedure. Some status codes displayed on the front panel are interpreted in conjunction with codes displayed on the processor LEDs (186 boards) or the display window (386 boards) on the rear of a processor.

If the unit is operating correctly, the status display code sequence stops at 20, indicating that the booting procedure is complete and the system is ready to run. If codes above 20 are displayed, a failure has prevented the system from booting properly.

The booting process normally takes about a minute. While waiting for a bootable disk to become ready, the status display alternates between A1 and 04.

Display codes 39 through 45 indicate that a failure has occurred during a processor booting operation. The code indicates the type of processor that failed. To further isolate the type of failure associated with these status display error codes, observe the processor LEDs on the processor board.

If code DE alternates with 41, 42, 43, 44, or 45, the master processor has not received a response from processors in expansion enclosures.

### General Processor LED Display

The status display on a General Processor board is a seven-segment LED that displays a single digit. During normal operation, one segment of the display is lighted. You can interpret disk activity from the position of the lighted segment, as described in Table B-1.

**Table B-1. SRP General Processor Display Activity**

| <b>LED Position</b>  | <b>Activity</b>                 |
|----------------------|---------------------------------|
| Top horizontal       | Disk seek in progress           |
| Middle horizontal    | Disk DMA in progress            |
| Bottom horizontal    | File system request in progress |
| Lower left vertical  | CPU busy                        |
| Lower right vertical | Termination                     |
| Decimal point        | Heartbeat                       |

### 186 Processor Board LED Display

During normal processing, you can use the LEDs on the rear of the processor to interpret what the processor board is doing, as listed in Table B-2.

**Table B-2. SRP 186 Processor Activity**

| LED Number | Activity                                               |
|------------|--------------------------------------------------------|
| 7          | System error (Reserved)                                |
| 6          | File system 2 (Seek in progress)                       |
| 5          | File system 1 (Direct memory access (DMA) in progress) |
| 4          | File system 0 (File system request in progress)        |
| 3          | CPU busy                                               |
| 2          | --                                                     |
| 1          | Termination                                            |
| 0          | Heartbeat                                              |

---

## Front Panel Status Codes

The status codes in this section appear on the front panel of the shared resource processor during system startup. Codes 0 through 20 do not represent an error condition.

- 0** The system is in the reset mode (keyswitch is in the Stop position).
- 01** Starting boot procedure; performing boot ROM diagnostics.
- 02** Boot ROM writing memory to *[Sys]<Sys>CrashDump.sys*.
- 03** Boot ROM performing full memory test (write/read/compare).
- 04** Boot ROM booting the operating system, *[Sys]<Sys>SysImage.sys*, on the master processor. Display alternates between 04 and A1 while waiting for system disk to become ready.
- 05** Boot ROM sequence successfully completed.
- 06** Hardware initialization begun.
- 07** Hardware initialization complete. First time master processor communicates across system bus.

## Shared Resource Processor Front Panel and Processor Status Codes

---

- 08 Master processor initialization complete.
- 09 Master processor initializes disk hardware.
- 10 Master processor reads master configuration file, *[Sys]<Sys>SrpConfig.sys*, and boots other processors.
- 20 System is operating normally.

### Boot Medium Error Codes

- 21 The master processor ROM was unable to find a bootable system image.
- 23 The master processor ROM found no valid home block on the first ready disk drive.
- 24 The master processor ROM found a bad run file signature on the first ready disk drive.
- 25 The run file checksum of the server operating system, *[Sys]<Sys>SysImage.sys*, on the first ready disk is bad, indicating that the file is corrupted.
- 26 Error when trying to reset.
- 27 Error when trying to rewind the tape.
- 28 Error when reading the tape.
- 29 Tape was written with incorrect or unsupported blocking factor.

### Disk Hardware Error Codes

- 30 Disk error on drive 0.
- 31 Disk error on drive 1.
- 32 Disk error on drive 2.
- 33 Disk error on drive 3.

- 36 Bad disk controller; ROM could not read the controller register.
- 37 Bad disk controller; not ready in command phase.
- 38 Bad disk controller; no response.
- 39 General board hardware error. A processor has reported a hardware error; see processor's rear panel LEDs for specific error.

### Processor Error Codes

- 40 Master processor detected FP time-out, indicating processor crash.
- 41 Master processor detected TP time-out, indicating processor crash. This code may alternate with code DE. See code DE for explanation and recommended action.
- 42 Master processor detected CP time-out, indicating processor crash. This code may alternate with code DE. See code DE for explanation and recommended action.
- 43 Master processor detected SP time-out, indicating processor crash. This code may alternate with code DE. See code DE for explanation and recommended action.
- 44 Master processor detected DP time-out, indicating processor crash. This code may alternate with code DE. See code DE for explanation and recommended action.
- 45 Master processor detected processor time-out, indicating processor crash. Processor type cannot be determined. This code may alternate with code DE. See code DE for explanation and recommended action.

### Software Error Codes

- 50 Fatal software error detected. See processor rear panel LEDs for error code.
- 51 Fatal software error detected. See the system log for more information. Try rebooting using a different configured keyswitch.

### Expansion Enclosure Error Codes

**DE** There is a break in the system bus between enclosures due to one of the following conditions: (1) An expansion enclosure is not turned on; (2) there is a faulty bus repeater board; (3) bus repeater cables are improperly installed or damaged.

If code DE alternates with code 41, 42, 43, 44, or 45, check to see whether:

- An expansion enclosure is not turned on
- A bus repeater board or the cabling between the bus repeater boards is faulty

Make sure all expansion enclosures are powered on.

If all expansion enclosures are powered on and code DE is still displayed, contact your customer service representative.

**E0** This value alternates with the enclosure number to indicate that power for that enclosure has not been applied. The first B enclosure is enclosure 2.

### Disk Hardware Error Codes

These codes also appear on the front panel status display.

**30** Disk error on drive 0.

**31** Disk error on drive 1.

**32** Disk error on drive 2.

**33** Disk error on drive 3.

### 186-Board Processor LED Codes

The following codes are derived from interpretation of the LEDs on the rear of 186 processor boards. See Section 19, "Troubleshooting," for information about interpreting the patterns that are displayed.



### Processor LED Codes While Front Panel Displays 01

These codes do not indicate an error condition. They can be used as an aid to diagnose problems. The LEDs should cycle through these numbers but not stop at them. An error condition occurs when the numbers are fixed in any of these values for any length of time.

- 01 Starting board hardware initialization.
- 02 Performing memory test of first 64K bytes of memory.
- 03 Setting up initial memory area.
- 04 Determining memory size.
- 05 Relocating CTOS/XE buffer. Setting up CDT.
- 06 Checking I/O ports.
- 07 Performing ECC wash to set up check bits.
- 10 Hardware initialization successfully completed.

### Processor LED Codes While Front Panel Displays 21

The following status codes describe problems with bad media.

- 21 Cannot boot from any disk.
- 22 No tape or disk device contains a valid server operating system, *[Sys]<Sys>SysImage.sys*.
- 23 No volume home block found on first ready drive.
- 24 Bad run file signature on first ready drive.
- 25 Bad run file checksum on first ready drive.

**Processor LED Codes While Front Panel Displays 39**

- 0A**      Bad ROM checksum.
- 0B**      Bad I/O port.
- 0C**      Memory error in local memory while writing 0's (zeros).
- 0D**      Memory error in local memory while writing 1's (ones).
- 0E**      Memory error in local memory while writing address pattern.
- 0F**      Memory error in local memory while reading address pattern.

The following codes describe problems with the disk hardware.

- 36**      Bad disk controller; ROM cannot read controller registers.
- 37**      Bad disk controller; never ready in command phase.
- 38**      Bad disk controller; no response.

The following error codes describe disk hardware error conditions on systems with a DP master processor.

- 3A**      Storage Processor (SP) of master DP does not have a Storage Controller (SC) board attached.
- 3B**      Inconsistent values on SMD controller interface.
- 3C**      SMD controller data input/output (DIO) bit set incorrectly.
- 3D**      A fatal SMD I/O error has occurred. (This is a generic error.)

The following error codes are displayed on ME boards.

- 4C**      Memory error on ME board while reading 0s.
- 4D**      Memory error on ME board while reading 1s.
- 4E**      Memory error on ME board while writing address pattern.
- 4F**      Memory error on ME board while reading address pattern.

### General Status Codes

- 1A** Error during hardware initialization.
- 1B** Error during data communications (data comm) peripheral initialization.
- 1C** Starting data comm dump or boot.
- 1D** Writing data comm dump.
- 1E** Reading data comm boot.
- 1F** Waiting for data comm line ACKnowledge.
- 81** Starting hardware initialization.
- 82** Starting checksum of ROM.
- 84** ECC logic and memory are initialized nondestructively (data is not lost).
- FF** Processor is in reset state.

### 386-Board Processor LED Codes

The following codes are derived from interpretation of the seven-segment display on the rear of 386 General Processor boards. See Section 19, "Troubleshooting," for information about interpreting the patterns that are displayed.

### Processor LED Codes While Front Panel Displays 01

These codes do not indicate an error condition. They can be used as an aid to diagnose problems. The display should cycle through these numbers but not stop at them. An error condition occurs when the numbers are fixed in any of these values for any length of time.

- 1 Starting board hardware initialization; performing ROM checksum.
- 2 Performing initial memory test of first 64K bytes of ROM.
- 3 Setting up initial memory area.
- 4 Determining board type; saving operating system structures; relocating operating system save buffer.
- 5 Setting up CPU definition table; checking I/O ports.
- 6 Performing ECC wash to set check bits correctly.
- 7 Check for operating system crash; determine if master processor.
- 8 Initialize RS-232-C channel B only; check for ampersand on B only.
- 9 Memory test if operating system did not crash.
- A Boot from QIC tape.
- B Boot from SCSI.
- C Boot from comm.
- D Dump to SCSI.
- F Nonmaster wait for boot.
- R Relocating the operating system.

This is the last step before the operating system takes control.

### Processor LED Codes While Front Panel Displays 39

Error conditions 1 through 6 are fatal. An E appears, followed by the number. This sequence is repeated until the problem is corrected.

- 1       Bad ROM checksum.
- 2       Bad I/O port.  
  
The boot code could not successfully write and read back port 140 (remote slot) and port 148 (Base Register 0).
- 3       Memory error in local memory while reading and writing 0s.
- 4       Memory error in local memory while reading and writing 1s.
- 5       Memory error in local memory while reading and writing addresses.
- 6       Memory error in local memory in local memory while verifying addresses.

Error conditions 7 through C appear only on the rear LEDs. An E appears, followed by the number. This sequence is repeated four times.

- 7       Bad runfile checksum.
- 8       Bad SCSI chip.
- 9       SCSI protocol error.
- A       RS-232-C boot error.
- B       Boot from QICI error.
- C       Run file header error.

### Processor LED Codes During Booting and Crash-Dumping

These codes do not indicate an error condition. They can be used as an aid to diagnose problems. The display should cycle through these numbers but not stop at them. An error condition occurs when the numbers are fixed in any of these values for any length of time.

- a Boot from QIC tape.
- b Boot from SCSI.
- c Boot from RS-232-C.
- d Dump to SCSI.
- f Unable to boot after first pass of devices. (21 is displayed on the front status panel.) After a few seconds, the ROM will go through the boot procedure again:
  - If the problem was in a SCSI disk boot process, the system will try again to boot from the SCSI disk.
  - If the system files were missing or corrupted and you insert a system tape, the system will boot from the system tape.
  - If the boot ROM would not wait for the system device to become ready, the system will try again. The device may respond on the second pass. (Note that this is not a frequent cause of this error condition.)
- r Relocating operating system in memory.

### System Crash Status Words

System crash status words can help you isolate the cause of a system crash. They are normally displayed on screen and are entered in the system log after a system crash.

The words are numbered 1 through 8, and their values are in hexadecimal. Depending on the corresponding error code, certain status words are either unused or dependent on the error code. See Table B-3 for a list of the status words and their meanings.



## Shared Resource Processor Front Panel and Processor Status Codes

---

The contents of the general status register (GSR) can also help you isolate the cause of a system crash. The contents of the GSR are reported to status word 3 when a system crash occurs. See Table B-4 for a list of the bits in the GSR and their meanings.

As an example, suppose the following error code and status words are reported as a result of a system crash:

```
SYSTEM CRASH - (ERC = 22)
```

```
Crash Information: 0016h 0008h D543h 0000h 0075h 0000h  
0301h 0009h
```

The status code indicates Error 22, a bus time-out. The information contained in the status words are as follows:

- Word 1 shows 16h, the equivalent of decimal 22.
- Word 2 indicates that the failure occurred in process 8.
- Bit 10 of word 3 indicates a bus time-out.

Bit 6 indicates that the trapped address was generated by the central processing unit (CPU) local to the processor where the error was detected.

Bits 1 and 0 indicate that the trapped address was greater than or equal to C0000h.

- Word 5 indicates that the processor in slot 75 was being referenced.
- Word 6 indicates that a memory address lower than FFFFh (64 KB) was being referenced.
- Words 7 and 8 indicate that the instruction preceding the instruction at address 03010:0009 was being executed when the time-out occurred.

The possible causes for the error could be isolated to a hardware failure of the processor in slot 74 or 75, or to a programming error.

## Shared Resource Processor Front Panel and Processor Status Codes

---

**Table B-3. System Crash Status Words**

| <b>Word</b> | <b>Error Codes</b>         | <b>Meaning</b>                                                                                                                                                          |
|-------------|----------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1           | All                        | The hexadecimal value for the corresponding decimal error code.                                                                                                         |
| 2           | All                        | The process number that was running when the error occurred.                                                                                                            |
| 3           | 21, 22, 23, 24, 25, 26, 28 | The 16-bit nonmaskable interrupt (NMI) status (read from the General Status Register, port hexadecimal 68).                                                             |
| 4           | 21, 22, 23, 24, 25, 28     | The high two bits of the memory address at the time of the NMI. If the value is 3, a remote memory reference was being performed when the crash occurred.               |
|             | 26                         | The interrupt service register.                                                                                                                                         |
| 5           | 22, 28                     | The contents of the remote slot number register (port hexadecimal 40). This value represents the slot number of the processor being referenced when the crash occurred. |
| 6           | 22                         | The contents of the base register zero (port hex 48). The base register zero holds bits 16 through 31 of the off-board memory address.                                  |
|             | 23, 28                     | The error correction code (ECC) syndrome register contents.                                                                                                             |
|             | 26                         | The interrupt request register.                                                                                                                                         |

continued

## Shared Resource Processor Front Panel and Processor Status Codes

---

**Table B-3. System Crash Status Words (cont.)**

| Word | Error Codes     | Meaning                                                                                                                                                                                                                                                                                                                                                                          |
|------|-----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7    | All (except 26) | The code segment (CS) of the location following the call to the fatal error handler.                                                                                                                                                                                                                                                                                             |
|      | 26              | The interrupt mask register.                                                                                                                                                                                                                                                                                                                                                     |
| 8    | All             | <p>The instruction pointer (IP) of the location following the call to the fatal error handler. Status words 7 and 8 indicate the memory location of the instruction immediately following the instruction that was being executed when the error occurred.</p> <p>The value in the CS register, shifted left one digit, plus the value in the IP register gives the address.</p> |

**Table B-4. General Status Register Contents**

| Bit | Name | Bit Setting | Meaning                                                                                                                                                                                             |
|-----|------|-------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 15  | ME   | 0           | Memory Expansion board is installed.                                                                                                                                                                |
| 14  | SE   | 0           | At least one processor in the system is driving the system error line on the system bus.                                                                                                            |
| 13  | LR   | 0           | Last reset by power up or RESET switch.                                                                                                                                                             |
|     |      | 1           | Last reset by software.                                                                                                                                                                             |
| 12  | NEM  | 0           | If an NMI was caused by a reference to a nonexistent memory location, the reference could have been initiated locally, or by a processor or direct memory access (DMA) controller on another board. |
| 11  | PF   | 0           | Power failure.                                                                                                                                                                                      |

continued

## Shared Resource Processor Front Panel and Processor Status Codes

---

**Table B-4. General Status Register Contents (cont.)**

| Bit | Name | Bit Setting | Meaning                                                                                                                                                                                                                                                                                                                       |
|-----|------|-------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 10  | TO   | 0           | Bus time-out. No acknowledge was received from a memory or I/O timing circuit.                                                                                                                                                                                                                                                |
| 9   | BE   | 0           | Bus error. A remote double-bit error or a reference to a nonexistent memory location occurred.                                                                                                                                                                                                                                |
| 8   | DBE  | 0           | Double-bit error.                                                                                                                                                                                                                                                                                                             |
| 7   | RME  | 0           | The trapped address was generated by another processor.                                                                                                                                                                                                                                                                       |
| 6   | CAD  | 1           | The trapped address was generated by the local CPU.                                                                                                                                                                                                                                                                           |
| 5   | DMA  | 1           | The trapped address was generated by the local DMA controller.                                                                                                                                                                                                                                                                |
| 4   | REM  | 1           | Front panel keyswitch is in the REMOTE position. Only applicable for the master processor.                                                                                                                                                                                                                                    |
| 3   | DFP  | 1           | This board is the master processor.                                                                                                                                                                                                                                                                                           |
| 2   | NOR  | 1           | Front panel keyswitch is in the NORMAL position. Only applicable for the master processor.                                                                                                                                                                                                                                    |
| 1   | A19  | 1           | Bit 19 of the trapped address. Used with bit 0 (A18) to determine if trapped address is located in base memory, expansion memory, or on another processor.                                                                                                                                                                    |
| 0   | A18  | 0           | Bit 18 of the trapped address.<br><br>If bits 19 and 18 are both 0, the trapped address is located in base memory, expansion memory, or on another processor.<br><br>If bits 19 and 18 are both 1, the trapped address is located on another processor.<br><br>Otherwise, the trapped address is located in expansion memory. |

---

# Appendix C

## Configuring Standard VGA

### Customizing the Display for Character-Based Applications

If you have an EISA- or ISA-bus workstation that does not include a CTOS VideoCard and default attribute mapping does not work for your application, you can customize the attribute mapping to something more appropriate for your needs. Four simple Executive commands can be used to specify alternatives for color, bold, struck-through, and underlined text. If you need to specify alternative displays for combinations of attributes, you can use a more powerful utility, the **VGA Character Attribute Mapper**. It maps each CTOS attribute combination (actually, the byte the application passes to VAM) to a specific combination of VGA attributes.

All these utilities modify a resource that defines how specific attribute combinations (attribute bytes) should be interpreted (displayed). The default mapping used by VAM itself is defined in a resource file, *[Sys]<Sys>VdmAttributes.res*. That resource file is a text file very much like a configuration file. The VAM resource file provides global attribute mapping on a system level.

A resource can also be included in the application run file itself; this provides application-specific attribute mapping. When an application run file contains a resource that specifies video-attribute mapping, VAM uses that mapping for that specific application, rather than the mapping specified in the VAM resource file. Version 4 run files (real mode) cannot include resources.

The VAM utilities can be used either to modify the VAM resource file or to add a mapping resource to the application run file. The **Resource Librarian** command can be used, as with any other resource, if you ever want to copy a resource from an existing run file. The **Resource Librarian** is distributed with the CTOS Development Utilities.

*Note: It is recommended that you modify the system resource file, rather than including resources with an application run file. If, however, you have two or more applications with very different needs, you may need to include the mapping with the run file itself.*

### Making Simple Modifications With Executive Commands

This subsection describes how to remap default attribute assignments with the following Executive commands:

**VGA Add Color Assignment**

**VGA Add Struck-Through**

**VGA Bold Replaces Underline**

**VGA Underline Replaces Bold**

By default, the Standard VGA hardware is programmed to support seven foreground colors plus black, seven background colors plus black, underlining, and blinking. CTOS supports eight colors (no black). The default mapping for Standard VGA, therefore, maps CTOS color index 2 and 3 to the same VGA color (that is, the fourth CTOS color, which is color index 3, is mapped to the third CTOS color, color index 2). The complete color mapping is described in more detail later in this appendix.

### Changing Color Mapping

Since the default color mapping maps the third and fourth CTOS colors (color indexes 2 and 3) to the same color, you may find that you need to change the color mapping for your application. To do so, you use the **VGA Add Color Assignment** command.



Color attributes are mapped by color index numbers 0 to 7. CTOS has a programmable color palette. Since the SignOn program, the Executive, and other applications can be set to change the color palette, you may specify only color indexes, and may not specify the color you see on the screen (for example, red). You may need to experiment to get the effect you want.

The following example shows how to change the default color assignment for a CTOS color index:

```
VGA Add Color Assignment
  [Run or resource file]           App.run
  From CTOS color index (0-7)     3
  To VGA foreground color index (1-7) 5
```

See the *CTOS Executive Reference Manual* for more detailed information about this command.

### Mapping Struck-Through to a Color

If your application uses the struck-through attribute, you may also want to map that to color. To do so, use the **VGA Add Struck-Through** command, as shown in the following example:

```
VGA Add Struck-Through
  [Run or resource file]           App.run
  [Background color index (0-7)]  7
```

### Substituting Bold for Underlining

Underlining is implemented using the Standard VGA alternate font. Bold can be displayed only if it is substituted for underlining as the alternate font.

## Configuring Standard VGA

---

Since the default mapping does not display the bold attribute, you might want to substitute bold for underlining. To do so, you use the **VGA Bold Replaces Underline** command. When you use this command, bold becomes the alternate font. That means that underlining no longer displays as underlining, so you can remap underlining to blinking, a color, or both, as shown in the following example:

```
VGA Bold Replaces Underline
[Run or resource file]      App.run
[Underline to use blinking?] Y
[Underline to use bold?]   N
[Background color index (0-7)] 5
```

Alternatively, you can choose to map both the underline and the bold attributes to bold text, as shown in the following example:

```
VGA Bold Replaces Underline
[Run or resource file]      App.run
[Underline to use blinking?] N
[Underline to use bold?]   Y
[Background color index (0-7)]
```

You use the **VGA Underline Replaces Bold** command to reverse the changes you made and to set underlining as the alternate font. You can also use that command to change the manner in which bold is displayed.

See the *CTOS Executive Reference Manual* for more detailed information about these commands.

## Displaying Current Mapping Attributes

If you want to check the current mapping for any file you can use the **VGA Display Mapping** command as shown below:

```
VGA Display Mapping
[Run or resource file] [Sys]<Sys>App.run
```

The mapping for the file is displayed as in the example below:

```
Global attribute mappings for this file:
CTOS Underline attribute supported.
Bold mapped to Blinking, color index 4
```

## Making More Extensive Modifications

This section describes attribute mapping theory and helps you to develop a more complicated mapping scheme, if the methods described earlier do not work for your applications. This typically occurs when you have an application that uses several combinations of CTOS attributes to display information to the user.

This section also briefly describes the **VGA Character Attribute Mapper** command. See the *CTOS Executive Reference Manual* for detailed information about that command.

### Attribute Mapping Theory

The *CTOS Programming Guide* describes how VAM deals with the translation from CTOS video attributes to Standard VGA attributes. With Standard VGA you can specify either underlining or bold to be used as the alternate font, but not both. If your application makes extensive use of both bold and underlining, you may need to use the **VGA Character Attribute Mapper** to map those attribute combinations (as attribute bytes) to other specific VGA attribute combinations.

CTOS supports the use of eight colors, while Standard VGA supports the use of seven colors plus black (that is, no color). These colors are stored in the system color palette and are not affected by attributes that are displayed. References to color in this document, therefore, are to an entry in the system color-palette (a color index) rather than to a specific color such as red or lavender.

You can use the **VGA Character Attribute Mapper** to remap colors. You will most often want to do this when your application uses both the third and fourth CTOS color palette entries (color indexes 2 and 3) at the same time to distinguish information. The **VGA Character Attribute Mapper** attempts to ensure that all attributes remain displayable (for example, by preventing a foreground color with the same background color).

## Configuring Standard VGA

---

Colors are selected from the system color palette by setting different patterns of color bits. Table C-1 shows how color-bit combinations select color indexes.

**Table C-1. Color-Bit Combinations**

---

| Color Bit 3 | Color Bit 2 | Color Bit 1 | Color Index |
|-------------|-------------|-------------|-------------|
| OFF         | OFF         | OFF         | 0           |
| OFF         | OFF         | ON          | 1           |
| OFF         | ON          | OFF         | 2           |
| OFF         | ON          | ON          | 3           |
| ON          | OFF         | OFF         | 4           |
| ON          | OFF         | ON          | 5           |
| ON          | ON          | OFF         | 6           |
| ON          | ON          | ON          | 7           |

---

Color indexes refer to the offset from the start of the list of colors in the color palette. For example, the first color palette entry has color index 0, the second color palette entry has color index 1, and the eighth color palette entry has color index 7. Note that the color index value is supplied to the **VGA Add Color Assignment** command described above, rather than the number listed in Table C-1.

CTOS systems allow applications to program each of the eight color-palette entries with any color (for example, red or blue). On Standard VGA systems, only seven of the eight color-palette entries can be programmed, because the first entry (color index 0) is always black (no color). On such systems, VAM ignores the fourth color palette entry (color index 3) when the palette is programmed. The default attribute mapping provides that colors displayed using the fourth palette entry are displayed using the same color as the third palette entry (color index 2). This mapping can be changed by using the **VGA Character Attribute Mapper**. Table C-2 lists the default mapping of CTOS color indexes to Standard VGA color indexes.

**Table C-2. Color Mapping: CTOS to Standard VGA**

| CTOS Color Index | Standard VGA Color Index |
|------------------|--------------------------|
| 0                | 1                        |
| 1                | 2                        |
| 2                | 3                        |
| 3                | 3                        |
| 4                | 4                        |
| 5                | 5                        |
| 6                | 6                        |
| 7                | 7                        |

CTOS VAM supports combining colors with the reverse-video attribute. When the reverse-video attribute is on, characters are displayed in black on a field of the indicated color. On Standard VGA, systems there is no notion of reverse-video. Rather, characters are displayed using foreground and background colors. Reverse-video is achieved by setting the foreground color to black and the background color to the CTOS color (according to Table C-2). When reverse-video is off, the background color is set to black and the foreground color is set to the CTOS color.

To map a CTOS attribute to a Standard VGA attribute that closely resembles the original CTOS attribute, you can use the following approach:

1. Copy the blinking and the underline attributes, if on, to the corresponding Standard VGA attributes.
2. If the reverse-video attribute is off, copy the color bits to the corresponding Standard VGA foreground color bits (according to the table above).
3. If the reverse-video attribute is on, copy the color bits to the corresponding Standard VGA background color bits (according to the table above.)
4. Add background color (or foreground color, if reverse-video is on) to substitute for the bold or struck-through attributes.

# Appendix D

## CTOS I/O Processor Card Error Codes

### Introduction

CTOS I/O Processor cards include basic diagnostic routines in the firmware that are run whenever the cards are booted. If a problem occurs error codes are passed to CTOS and are written to the system log file. You can view the contents of the log file by using the **Plog** command.

The error codes are listed here. Since corrective action for many of the codes is the same for the user or customer service representative, the codes are listed in ranges and individual explanations for each are not provided here.

The error codes are listed in decimal format.



### Error Codes

- 00  
to  
15       Reserved.
- 16  
to  
31       A fatal error has occurred that prevents the IOP card from booting.  
Replace the IOP.
- 32  
to  
47       An error has occurred that affects the firmware running on the IOP  
card.  
Check the memory on the card or replace the card.
- 48  
to  
63       An error has occurred with the Cluster IOP card.  
Replace the IOP card.
- 64       Reserved.
- 65  
to  
69       An error has occurred with the Communications IOP card.  
Replace the IOP card.
- 70  
to  
79       Reserved.
- 80  
to  
96       An error has occurred with the Ethernet IOP card.  
Replace the IOP card.

97

to

128

Reserved.

129

Memory bank 0 is bad or missing.

This error indicates mismatched memory configuration or memory failure.

Check your memory configuration using the SCU first. If no problem seems to exist, try replacing the SIMM.

130

Memory bank 1 is bad or missing.

This error indicates mismatched memory configuration or memory failure.

Check your memory configuration using the SCU first. If no problem seems to exist, try replacing the SIMM.

131

Memory bank 0+1 is bad or missing.

This error indicates mismatched memory configuration or memory failure.

Check your memory configuration using the SCU first. If no problem seems to exist, try replacing the SIMM.

132

Memory bank 2 is bad or missing.

This error indicates mismatched memory configuration or memory failure.

Check your memory configuration using the SCU first. If no problem seems to exist, try replacing the SIMM.

133

Memory bank 0+2 is bad or missing.

This error indicates mismatched memory configuration or memory failure.

Check your memory configuration using the SCU first. If no problem seems to exist, try replacing the SIMM.

## CTOS I/O Processor Card Error Codes

---

- 134** Memory bank 1+2 is bad or missing.  
This error indicates mismatched memory configuration or memory failure.  
Check your memory configuration using the SCU first. If no problem seems to exist, try replacing the SIMM.
- 135** Memory bank 0+1+2 is bad or missing.  
This error indicates mismatched memory configuration or memory failure.  
Check your memory configuration using the SCU first. If no problem seems to exist, try replacing the SIMM.
- 136** Memory bank 3 is bad or missing.  
This error indicates mismatched memory configuration or memory failure.  
Check your memory configuration using the SCU first. If no problem seems to exist, try replacing the SIMM.
- 137** Memory bank 0+3 is bad or missing.  
This error indicates mismatched memory configuration or memory failure.  
Check your memory configuration using the SCU first. If no problem seems to exist, try replacing the SIMM.
- 138** Memory bank 1+3 is bad or missing.  
This error indicates mismatched memory configuration or memory failure.  
Check your memory configuration using the SCU first. If no problem seems to exist, try replacing the SIMM.
- 139** Memory bank 0+1+3 is bad or missing.  
This error indicates mismatched memory configuration or memory failure.  
Check your memory configuration using the SCU first. If no problem seems to exist, try replacing the SIMM.

- 140** Memory bank 2+3 is bad or missing.  
This error indicates mismatched memory configuration or memory failure.  
Check your memory configuration using the SCU first. If no problem seems to exist, try replacing the SIMM.
- 141** Memory bank 0+2+3 is bad or missing.  
This error indicates mismatched memory configuration or memory failure.  
Check your memory configuration using the SCU first. If no problem seems to exist, try replacing the SIMM.
- 142** Memory bank 1+2+3 is bad or missing.  
This error indicates mismatched memory configuration or memory failure.  
Check your memory configuration using the SCU first. If no problem seems to exist, try replacing the SIMM.
- 143** Memory bank 1+2+3+4 is bad or missing.  
This error indicates mismatched memory configuration or memory failure.  
Check your memory configuration using the SCU first. If no problem seems to exist, try replacing the SIMM.
- 144**  
to  
**157** A problem has occurred in communications between the firmware and the operating system, at the firmware level.  
Try replacing the IOP card. If this does not work, try a new copy of *[Sys]<Sys>Sysimage.sys*
- 160**  
to  
**174** An error has occurred with the ISDN IOP card.  
Replace the IOP card.



# Glossary

^

See circumflex.

## **\$ directory**

See dollar-sign directory.

## **A**

### **agent cache**

Enables caching of files from the server to the client workstation's local cache. It also enables caching on Cluster IOPs of multiprocessor workstations.

### **application**

A program or software package that accomplishes a specific task or set of tasks, such as office publishing, spreadsheet preparation, or database retrieval.

### **archive dataset**

A file or set of files that contains the contents of other files in a compressed form. An archive dataset is created by the **Volume Archive** or **Selective Archive** command.

### **archive media**

Floppy diskettes, hard disks, or tapes that are used to store backups.

### **active command file**

The command file specified in the user configuration file with which the user signed on. If none is specified, *[Sys]<Sys>Sys.cmds* is the default.

### **active password**

The password that is in effect when a command is issued.



### **ASCII text file**

A file containing the alphanumeric characters comprising the American Standard Code for Information Interchange. This term is commonly used to denote a file containing unformatted text, as opposed to a file containing imbedded formatting characters, which may not be visible to the user.

### **asynchronous terminal emulator (ATE)**

A workstation functioning as an asynchronous terminal. Also used to describe a software package that provides this capability. *See also* Basic ATE.

## **B**

### **bad block table**

An area on a disk where the addresses of bad spots are stored. Data is not written to the bad spots identified in the bad block table.

### **Basic ATE**

A software product that allows a workstation to function as an asynchronous terminal. *See also* asynchronous terminal emulator.

### **baud rate**

The speed at which data is transmitted.

### **binary file**

A file containing a set of computer instructions reduced to a choice of two alternative conditions.

### **bit**

The smallest unit of electronic data.

### **BIOS**

Acronym for basic input/output system, BIOS is a firmware program that is present on EISA/ISA systems.

### **BNet**

A communications application used to connect servers together. *See also* network node.

**bootstrap (also boot)**

The hardware-initiated process of loading an operating system into memory.

**buffer**

An area of memory used as a temporary holding place for data.

**byte**

A unit of data containing a specific number of bits.

**C****cache**

A high-speed buffer that improves data access speed.

**cache memory disk**

An area of memory that is used as a disk, usually as the scratch volume.  
*See also* scratch volume.

**case sensitive**

Upper case and lower case letters are interpreted differently. The Executive is not case sensitive.

**channel**

A path along which signals can be sent, for example, a communications channel or an output channel.

**circumflex ( ^ )**

Precedes a password in a file specification.

**click**

To press and then immediately release a mouse button.

**client workstation**

A workstation that is connected to a server. Formerly called *cluster workstation*. *See also* server.

**cluster**

A group of workstations connected to a common server. *See also* server.

### **Cluster File Access (CFA)**

A method by which workstations can gain access to disks on other workstations in the cluster.

### **Cluster View**

A software product that connects the keyboard and monitor on a client workstation to a processor on the server.

### **cluster workstation**

*See* client workstation.

### **CMOS**

Acronym for complementary metal-oxide semiconductor. It refers to the battery-backed up RAM on EISA/ISA workstations.

### **command case**

An arbitrary value assigned within a run file that invokes a particular function of the program.

### **command file**

A file the Executive reads to display command names, command forms, and help descriptions, and to associate a run file and command case with each command.

### **command form**

Displays parameter fields for the specified command.

### **command line**

The highlighted line on the Executive screen, where you enter the name of the command you want to issue.

### **commented text**

Text that has no effect when a program is compiled or executed.

### **configuration**

An arrangement of parts, such as computer hardware, or of elements, such as software programs.

**configuration file**

Contains parameter values for a software product or hardware device.

**Context Manager**

A program that divides memory into multiple partitions so that more than one program can be started, and in some cases simultaneously executed, on a workstation. *See also* memory partition, static partition, and variable partition.

**corrupted volume**

An initialized disk that contains damaged or unreconciled data.

**CP**

Acronym for an SRP Cluster Processor.

**CPU**

*See* processor.

**crash**

*See* system crash.

**crash dump**

The process of writing data from memory into a file, so that it can be analyzed and debugged by an operating system engineer.

**CRC**

*See* cyclic redundancy check.

**CTOS**

A comprehensive term encompassing all varieties of the CTOS I, CTOS II, CTOS III, and CTOS/XE operating systems.

**CTOS I**

A set of real-mode operating systems for workstations.

**CTOS II**

A set of protected-mode operating systems for workstations.

**CTOS III**

A set of virtual-memory operating systems for workstations.

### **CTOS I/O Processor Card (IOP)**

An expansion card for EISA workstations that contains its own processor.

### **CTOS/XE**

A set of real-mode and protected-mode operating systems for shared resource processors.

### **cursor**

A movable marker that indicates where on the screen the next typed character will appear.

### **customized operating system**

An operating system that has been customized with the System Build Utilities. *See also* prebuilt operating system.

### **cyclic redundancy check (CRC)**

An error-checking procedure that takes place during cluster communications.

## **D**

### **Debugger**

A software debugging product used by programmers.

### **default directory**

The directory name that appears in angle brackets (< >) in the path setting on the screen.

### **default path**

The volume and directory that appear in the path setting on the screen. *See also* path.

### **default value**

A predetermined value with which a command is executed when an optional field is left blank.

### **default volume**

The volume or device name that appears in square brackets ([ ]) in the path setting on the screen.

**demand paging**

The ability of an operating system to swap portions of a program (pages) in and out of memory (frames) as they are needed. *See also* virtual memory.

**Development Utilities**

A software package containing commands and applications that are used primarily by programmers and software development engineers.

**device**

A disk drive, printer, tape drive, modem, or other physical device that receives or transmits data.

**device name**

*See* device specification.

**device password**

A password assigned to a piece of hardware. Device passwords are assigned by the operating system or in the operating system configuration file.

**device specification**

The identifier for a piece of hardware. Device specifications are assigned by the operating system or in the operating system configuration file.

**device template**

A set of configuration file entries that define the physical characteristics of a disk device. Such parameters are required to format a disk.

**digital data storage (DDS) drive**

A tape drive on a workstation that uses small, cassette-like tape cartridges.

**direct memory access (DMA)**

A direct high-speed data transfer between an input/output device and memory.

**directory**

A subdivision of disk storage space.



## Glossary

---

### **disk**

A mass-storage device for data.

### **disk drive**

The mechanism that holds the disk.

### **disk drive heads**

The mechanisms that read data from and write data to the disk.

### **diskless workstation**

Consists of a base unit containing the CPU and video controller. It uses disks on the server for data storage; disks are not connected directly to it.

### **distribution media**

The diskettes or tapes on which software is supplied.

### **DMA**

*See* direct memory access.

### **dollar-sign directory**

A directory that stores temporary files.

### **DP**

Acronym for an SRP Data Processor.

## **E**

### **ECC**

*See* error checking and correction.

### **Editor**

An ASCII text editing application.

### **EISA**

Acronym for extended industry standard architecture.

**EISA/ISA-bus workstation**

Consists of a base unit containing a CPU and other hardware components such as a hard disk, a floppy disk drive, and a video controller. It can be upgraded with "industry standard" hardware components. The SuperGen Series 3000 is an EISA/ISA-bus workstation.

**error checking and correction (ECC)**

Detection and correction of single-bit errors in the processing unit.

**error code**

A decimal or hexadecimal number denoting an error condition on a workstation or shared resource processor.

**Executive**

The CTOS command interpreter.

**F**

**field**

The highlighted line in a command form where a parameter value is entered.

**file**

A set of data that is stored and retrieved as a unit.

**file specification**

A unique identifier that contains the name of a file, as well as its volume and directory location.

**file system**

The data and control structures stored on accessible disks.

**file-system cache**

An area of memory where disk sectors are stored dynamically, as they are used.

**floppy disk**

A small, removable data storage disk.

### **floppy disk drive**

A slot-like opening on a workstation that holds a floppy disk.

### **format**

A particular arrangement of data.

### **format template**

A set of configuration file entries that define the characteristics of a volume. Such parameters are required to initialize a disk.

### **FP**

Acronym for an SRP File Processor.

### **fragmentation**

Noncontiguous data storage. A fragmented file is stored in multiple file extents. A fragmented disk contains many small, noncontiguous areas of storage space.

### **frame**

A 4K-byte region of physical memory into which the paging service loads a page of program code or data.

### **function keys**

The keys labeled **F1** through **F10**. Their functions change from program to program.

## **G**

### **G byte** (*also gigabyte*)

1,073,741,824 bytes (1,024M bytes).

### **Generic Print System (GPS)**

A set of software programs that provide printing services for CTOS applications.

### **global thrashing**

A condition that occurs when the number of frames required by concurrently running programs exceeds the amount of physical memory.

**GP**

Acronym for an SRP General Processor.

**H**

**half-inch tape drive**

A reel-to-reel tape drive for use on shared resource processors only.

**hexadecimal number**

A number in the base sixteen number system, which is primarily used by programmers. Hexadecimal digits are represented by numerals 0 to 9 and letters A, B, C, D, E, and F.

**I**

**I-bus**

The keyboard bus on traditional CTOS X-Bus workstations.

**IDE drive**

A type of disk drive that can be installed in EISA/ISA workstations.

**input/output (I/O)**

Data transfers between subsystem boundaries, such as from disk to memory, then back to disk.

**input/output processor (IOP)**

An expansion card for EISA workstations that contains its own processor.

**integrated X-Bus workstation**

Contains the processor, a hard disk, and a floppy disk drive in the base unit, and can be expanded with X-Bus modules. The B39 style is an integrated X-Bus workstation. *See also* X-Bus workstation.

**IOP**

*See* CTOS I/O Processor Card.

### ISA

Acronym for industry standard architecture.

### ISDN

Acronym for Integrated Services Digital Network.

## J

### Job Control Language (JCL)

A programming language processed by the Batch facility.

## K

### K byte (*also* kilobyte)

1,024 bytes.

### keyword

A predefined word or string of characters that identifies a parameter. Keywords are used in many configuration files and are written in the form of *:Keyword:*. The parameter value follows the closing colon.

## L

### LFS

*See* local file system.

### loadable request file

A binary file containing request definitions for a system service.

### local file system (LFS)

A workstation with its own disks, as opposed to a diskless workstation, which always uses disks on the server.

### logging out

The opposite of signing on. Logging out exits the Executive.

**M**

**M byte** (*also megabyte*)

1,048,576 bytes (1,024K bytes).

**master processor**

The first processor in the primary cabinet of an SRP. It bootstraps itself first and then controls booting of the other processors.

**memory**

High-speed volatile data storage, the contents of which can be altered at any time. *See also* random access memory.

**memory block**

Small portions of memory that are allocated by the operating system for use during interprocessor data transfers.

**memory disk**

A portion of memory on a shared resource processor that functions as a system volume.

**memory partition**

A discrete area of memory. *See also* Context Manager, static partition, and variable partition.

**message file**

A binary file containing the screen prompts and messages displayed by an application.

**message text file**

The text source file for a binary message file.

**modify access**

An access level that allows changes to be made to a file.

**module**

A workstation component, such as a disk drive, housed in its own casing and connected as an individual unit.



### **mouse**

An electronic pointing device, used for drawing or selecting items on the screen.

### **multiprocessor workstation**

An EISA workstation that contains one or more CTOS I/O processor cards.

## **N**

### **network node**

A server connected via BNet to other servers in a network. Client workstations connected to a node can communicate with other network nodes.

### **node**

See network node.

### **notebook workstation**

A small portable workstation that can be easily detached and then reconnected to the cluster.

### **null device**

A valid, but nonexistent, device specification [*Nul*]. It is used to test command execution without generating output.

## **O**

### **operating system**

A program that controls execution of other programs and coordinates use of peripherals on the computer.

### **operating system configuration file**

A file containing configurable operating system parameters.

### **output**

Data delivered from a program to a file or device.

**oversubscribed memory**

A condition where the total number of pages allocated to applications and system services is greater than the number of frames available. *See also* virtual memory.

**overwrite**

To replace the contents of an existing file with the contents of another file. Overwriting destroys the original file. This is an option for many Executive commands.

**P**

**page**

A 4K-byte section of a program in the linear address space.

**page fault**

The process of bringing pages (of data) into physical memory.

**paging service**

The virtual memory operating system service that controls demand paging.

**parameter**

A definable element of information affecting the way a program is executed.

**parameter field**

*See* field.

**parameter template**

A format or device template read by the **Format Disk** command. *See also* device template and format template.

**parameter value**

An element of information supplied in a command form or a configuration file.

**partition**

*See* memory partition.

### **partitioned memory management**

The memory management system whereby the operating system loads programs into individual memory partitions. *See also* demand paging and virtual memory.

### **password**

An access code that restricts the use of a system. Workstations or servers can have several passwords that allow varying levels of access to different users. *See also* active password.

### **path**

The default volume and directory. This volume and directory are used automatically when you execute a command unless you override the path with a file specification. The path setting appears in the status area of the screen.

### **physical address space**

The actual amount of memory that can be addressed by a program. In real mode, the physical address space is 1M byte. In protected mode, the physical address space is determined by the system processor and its hardware limitations.

### **prebuilt operating system**

An unmodified version of a CTOS operating system distributed by Unisys Corporation.

### **prefix files**

Operating system source code files that are modified, assembled, and linked to build a customized operating system.

### **primary file headers**

The file headers used by the operating system to perform disk read and write operations. *See also* secondary file headers.

### **primary partition**

The memory partition containing the program that is currently active on the workstation or shared resource processor.

**private installation**

Software that is installed for use on the local workstation only. *Compare to public installation.*

**processor** (*also* processing unit or CPU)

The unit that interprets and executes instructions.

**protected mode**

A program or operating system that can use memory above the first megabyte.

**protection level**

A number assigned to a file that designates read or modify access and whether a volume, directory, or file level password is required to open it.

**public installation**

Software that is installed on the server and can be accessed by other workstations in the cluster. *Compare to private installation.*

**Q**

**quarter-inch cartridge (QIC) tape drive**

A tape drive on a workstation or shared resource processor that uses cassette-like cartridge tapes.

**queue**

A portion of memory used for storing a list of files or jobs awaiting processing.

**Queue Manager**

A system service that controls spooled printing and other queue-oriented jobs, such as batch processing.

### R

#### **random access memory (RAM)**

A high-speed storage area where data is loaded prior to processing. The contents of memory are volatile and can be altered at any time. After processing, data is written back to disk for permanent storage.

#### **RDAT**

*See* digital data storage drive.

#### **read access**

An access level that allows a file to be opened or processed only, but not to be changed.

#### **real mode**

A program or operating system that runs in the first megabyte of memory.

#### **record**

To store a group of commands that can be reexecuted later.

#### **remote caching**

The caching of disk sectors to a file system cache located on a processor different from the one to which the disk is physically connected.

#### **Remote Keyboard Video Service (RKVS)**

The Cluster View system service.

#### **Remote User Manager (RUM)**

A memory manager that allows workstations to execute applications via Cluster View in discrete partitions on a shared resource processor.

#### **request file**

*See* loadable request file.

#### **RUM**

*See* Remote User Manager.

#### **run file**

An executable program.

**S****scratch volume**

A disk used for storage of the temporary files that some applications create.

**SCSI**

Acronym for Small Computer System Interface. It provides a design standard for hardware device interfaces.

**SCU**

See System Configuration Utility.

**secondary file headers**

Duplicate copies of primary file headers. They are used to retrieve data when primary file headers are damaged. *See also* primary file headers.

**sector**

512 bytes of data.

**server**

A workstation or shared resource processor (SRP) to which client workstations are connected. The server controls many system resources, such as printing and communications. Co-workers can share the files and applications that are stored on disks located on the server.

**Setup program**

The hardware configuration program on ISA workstations.

**shared resource processor (SRP)**

A multiprocessor computer that is always used as a server. Also called an *XE*.

**sign on**

The procedure that starts a user session. The user signs on with a predefined name, which determines the applications and commands that are available.



### **SMD disk drive**

Acronym for Storage Module Device. It refers to an eight-inch hard disk drive for use on a shared resource processor.

### **Software Release Announcement (SRA)**

A version-specific document containing information about a new release of a software product. Formerly called *Release Notes*, *Release Notice*, or *Release Information File*.

### **source code**

The text of a programming language, before it is compiled and linked to form an executable program.

### **SP**

Acronym for an SRP Storage Processor.

### **squash**

To reduce file and disk fragmentation.

### **SRP**

*See* shared resource processor.

### **ST-506 disk**

A non-SCSI SRP disk.

### **Standard Software**

A set of programs, configuration files, and commands packaged with the operating system, which are needed to configure the system and perform basic operations.

### **static partition**

An area of memory that is allocated by Context Manager to be a specific predetermined size. *See also* Context Manager, memory partition, and variable partition.

### **status area**

The top two lines of the Executive screen where the default path, user name, and date/time information are displayed.

**status code**

A number designating a certain condition on the system. In many cases, status codes represent errors. In other cases, they represent a normal operating condition.

**Sys**

An abbreviation for "system." It is used in file names to denote files that are necessary for the workstation or SRP to boot and function correctly.

<Sys>, when enclosed in angle brackets, denotes the name of a directory on every disk that contains unique information about the disk. This information is used by the operating system when you issue commands and use applications.

[Sys], when enclosed in square brackets, is the system-assigned pseudonym for the volume from which the workstation or shared resource processor bootstraps.

[Sys]<Sys>, when used together, denote the volume and directory containing the bootable operating system.

**System Build Utilities**

The set of commands and prefix files used to build a customized operating system.

**System Configuration Utility (SCU)**

The hardware configuration utility on EISA workstations.

**system crash**

An abnormal condition from which the system cannot recover. After a crash, the system usually freezes or reboots automatically.

**system error log**

A file containing information about many types of hardware and software errors. It can be displayed or printed with the **PLog** command.

**system image**

The operating system.

### **system service**

A program than expands the capabilities of the operating system.

### **system software**

The operating system, system services, utilities, and configurations files that are required for a workstation or shared resource processor to function as a computer.

## **T**

### **tape file mark**

A software mark that separates sequential tape files.

### **TeleCluster**

The method of connecting workstations to a server via telephone lines.

### **template file**

A text file containing menu displays and function key definitions for a particular command.

## **U**

### **user configuration file** (*also* user file)

A configuration file that defines one or more software environments for the user.

### **user name**

The name with which a user signs on to the system.

### **utility**

A program that carries out a specific task, such as copying or deleting files.

**V****valid volume**

A disk that has been formatted and initialized for use on a workstation or shared resource processor.

**value**

An element of information supplied in a command form or a configuration file.

**VAM**

See Video Access Method.

**variable**

A predefined character or group of characters that is replaced with an actual value during program execution.

**variable partition**

An area of memory that is allocated by Context Manager of a size equal to or smaller than a specified value. *See also* Context Manager, memory partition, and static partition.

**version number**

A number designating the revision level of a software product.

**Video Access Method (VAM)**

The video services software product that is installed separately from the operating system.

**virtual memory**

The apparent size of memory (to an application), which is greater than actual physical memory size, that makes it possible to address more memory than physically exists on the processor. *See also* demand paging.

**volume**

An initialized disk. *See also* valid volume.

### **volume control structures**

The framework within which the file system allocates disk space.

### **volume name**

The name assigned to a disk when it is initialized.

### **volume password**

Allows unrestricted access to a volume.

## **W**

### **W-block**

See memory block.

### **workstation**

A desktop computer that can function as a standalone system or be connected into a workgroup called a cluster. *See also* cluster.

### **workstation-type number (WsNNN)**

A three-digit number assigned to the operating system for each type of processor.

### **write access**

The capability of opening a file and making changes to it.

### **write-behind cache**

A file-system caching method that stores changes made to files in memory for an indefinite period of time before writing them to disk. *Compare to* write-through cache. *See also* file-system cache.

### **write-through cache**

A file-system caching method that writes changes made to files directly to disk. *Compare to* write-behind cache. *See also* file-system cache.

**X****X-block**

Memory that is allocated by the operating system for use during communications between the server and client workstations.

**X-Bus workstation**

A collection of separately housed modules, each containing an individual hardware component, such as a processor, a disk, or a graphics controller. X-Bus workstations include the B26, B28, B38, and B39 styles. *See also* integrated X-Bus workstation.

**X-Bus+ workstation**

Consists of a base unit containing the CPU and removable cartridges, which house disk drives, graphics controllers, tape drives, and other optional equipment. The SuperGen Series 5000 is an X-Bus+ workstation.

**XE**

*See* shared resource processor.

**Y****Y-block**

*See* memory block.

**Z****Z-block**

*See* memory block.





# Index

## A

- access modes, 6-5
- AccessService.run*, 6-14
- acronyms for processors, 2-10
- administrative duties, 1-3, 1-5
- Administrator Cluster View**
  - command, 4-12
  - Command Access Service, with, 6-14
- agent cache, 14-6
  - Cluster IOP, on, 14-7
  - diskless workstation, on, 14-7
- :AgentCacheDefaultEnable:*, 14-6, 14-7
- applications
  - definition of, 1-1, 8-2
  - disk-intensive, 15-9
  - installing, 8-3
  - removing, 8-12
  - troubleshooting, 8-14, 19-13, 19-14
- archive dataset, 13-1
- assembling prefix files, 17-7

## B

- B-box, 2-7
- B26 workstation, 2-2
- B28 workstation, 2-2
- B38 workstation, 2-2
- B39 workstation, 2-2

- backups
  - archive dataset, 13-1
  - Cluster View, with, 13-7
  - complete volume, 13-2
  - corrupted volumes, 13-14
  - incremental, 13-6
  - restoring, 13-8
  - routine, 13-1
- Bad Sector** command, 11-31
- bad spots, disks
  - file, 11-10, 11-11
  - non-SCSI format, 11-10
  - requirements for, 11-6
  - SCSI format, 11-11
- .bin* files, 18-2
- binary message files, 18-1, 18-3
- BIOS, 2-14
- BNet Cluster Access, 10-5
- :Boot:*, 16-6, 16-12
- boot ROM, 5-1, 5-2, 5-8, A-1 to A-2
  - BIOS, 2-14
- bootable media
  - floppy diskettes, 19-32
  - tape, 19-37
- Bootstrap** command, 5-7, 16-9, 17-10, 17-12
- Bootstrap menu, 5-5
  - EISA/ISA-bus workstation, 5-6
  - X-Bus workstation, 5-6
  - X-Bus+ workstation, 5-6
- bootstrap ROM, 5-1, 5-2, 5-8, A-1 to A-2

- BIOS, 2-14
- bootstrapping
  - ClientCard, 5-1, 5-3, A-1 to A-2, A-55
  - EISA/ISA-bus workstation, 5-1, 5-2, A-1 to A-2
  - floppy diskette, from, 19-32
  - hardware IDs, 5-7
  - indirect, 5-7
  - menu, 5-5
  - multiprocessor workstation, 5-2
  - server, from, 5-3, 19-31
  - SRP, 5-8
  - tape, from 19-37
  - workstation, 5-1
  - workstation-type numbers, 5-3, 5-4
- building an operating system
  - assembling, 17-7
  - build directory, 17-4
  - linking, 17-7
  - prefix files, 17-4
    - cluster agent, 17-8
    - CTOS I, 17-5
    - CTOS II, 17-6
    - CTOS III, 17-6
    - CTOS/XE, 17-7
  - file system, 17-7
  - modifying, 17-4
  - operating system, 17-9
  - required software, 17-2
  - rmSysGen.asm*, 17-3
  - SysGen.asm*, 17-4
  - testing, 17-10
  - troubleshooting
    - assembly errors, 17-13
    - bootstrap errors, 17-14
    - link errors, 17-13

## C

- C-box, 2-7
- cabinets, SRP, 2-6
  - primary, 2-7
  - summary of, 2-7
- cache memory disk, 15-12
- Cache Status** command, 14-2
- cache, file-system, 14-1. *See also* file-system cache.
- cartridges, X-Bus+, 2-5
- CD-ROM
  - device names, 12-7
  - drives, 12-6
  - software installation from, 8-10
  - system service, 9-2, 12-7
- CD-ROM Service, 9-2, 12-7
- CFA Configure** command, 10-2
- CFA Display Volume**
  - Information** command, 10-5
- Change Volume Name** command, 6-4
- cleaning up disks, 13-1
- client workstation, 1-1
- :ClientAgentFile:*, 16-13
- ClientCard, booting, 5-1, A-1 to A-2, A-55
- cluster
  - definition of, 1-1, 2-1
  - troubleshooting, 19-12
- Cluster Access system service, 10-5
- cluster agent, assembling and linking, 17-8
- Cluster.cmds*, 8-11
- Cluster File Access, 9-2, 10-1
  - configuring, 10-2
  - File Filter service, installing, 10-4
- floppy drives, 10-5

- Server Service, installing, 10-3
- system services, 10-1
- Workstation Agent, installing, 10-3
- Cluster IOP, file system cache, 14-7
- cluster local file system (LFS), 3-4
- Cluster Processor, description of, 2-9
- Cluster Status** command, 19-4
- Cluster View
  - administrator session, 4-12, 4-18
  - backups with, 13-7
  - commands, 4-12
  - exiting, 4-16
  - function keys, 4-16
  - installing system services with, 9-19
  - Login Service, 4-17
  - menu, 4-15
  - parameter fields, 4-14
  - Remote Keyboard Video Service (RKVS), 4-11
  - Remote User Manager (RUM), 4-11
  - single-processor server, using on, 4-16
  - SRP, using on, 4-10
  - starting a session, 4-13
  - system services, 4-11, 4-13, 9-19
  - user file options for, 7-20
  - videoless workstation server, using with, 6-13
  - working in a session, 4-15
  - workstation server, using on, 4-10
- Cluster View** command, 4-12
  - Command Access Service, with, 6-14
- `:ClusterLine1:` and `:ClusterLine2:`, 16-13
- CMOS, 2-14
- code sharing, 15-4
- color attributes, setting
  - Executive, 18-8
  - Standard VGA, C-2
- Command Access Service, 9-2
  - configuring, 6-15
  - installing, 6-14
  - log file, 6-16
  - remote nodes, accessing, 6-17
- command files, private and public, 8-11
- commands
  - limiting access to, 6-14
  - system services, for, 9-15
- commands, by name. *See also* the individual command name listings.
  - Administrator Cluster View**, 4-12, 6-14]
  - Bad Sector**, 11-31
  - Bootstrap**, 5-7, 16-9, 17-10, 17-12
  - Cache Status**, 14-2
  - CFA Configure**, 10-2
  - CFA Display Volume Information**, 10-5
  - Change Volume Name**, 6-4
  - Cluster Status**, 19-4
  - Cluster View**, 4-12, 6-14
  - Configure Disk**, 6-2, 6-3, 11-28
  - Configure Sequential Access Device**, 12-4
  - Create Directory**, 6-7
  - Create File**, 19-29
  - Create Message File**, 18-3
  - Disable Caching**, 14-4
  - Disable Cluster**, 13-2
  - Disk Partition Display Details**, 11-30
  - Disk Partition Set Active**, 11-29

- Editor**, 4-6
- Enable Caching**, 14-5
- FDISK (DOS)**, 11-30
- Format Disk**, 6-3
- Install**, 8-4
- Install CDROM Service**, 12-7
- Install CFA File Filter**, 10-4
- Install CFA Server Service**, 10-3
- Install CFA Workstation Agent**, 10-3
- Install Command Access Service**, 6-14
- Install MCR Service**, 7-13
- Install Sequential Access Service**, 12-3
- Installation Manager**, 8-3
- Link CTOS**, 17-10
- Link CTOS I**, 17-10
- Link CTOS II**, 17-10
- Link CTOS III**, 17-10
- Link CTOS VM**, 17-10
- List Message File**, 18-2
- Lock In Cache**, 14-6
- Merge Binary Message Files**, 18-4
- Merge Message Files**, 18-4
- Mini Customizer**, 17-2
- Multiprocessor Statistics**, 15-12
- Partition Status**, 9-11, 15-4, 15-5, 19-5
- PLog**, 11-31, 13-13, 19-1
- Read Hardware ID**, 5-7
- Restore Archive**, 13-9
- Screen Setup**, 18-8
- Set Directory Protection**, 6-8, 6-9
- Set Protection**, 6-10, 6-11, 6-12
- Set Time**, 6-14
- Software Installation**, 8-4
- System Manager**, 4-2
- Tape Erase**, 12-5
- Tape Retension**, 12-5
- Unlock Cache**, 14-6
- User File Editor**, 7-1
- Version**, 15-2
- VGA Add Color Assignment**, C-2
- VGA Add Struck-Through**, C-3
- VGA Bold Replaces Underline**, C-4
- VGA Character Attribute Mapper**, C-5
- VGA Display Mapping**, C-4
- VGA Underline Replaces Bold**, C-4
- Video**, 18-8
- Volume Archive**, 13-2
- Volume Status**, 11-33
- Write Hardware ID**, 5-7
- XESoftware Installation**, 8-4
- communications applications, 8-2
- Communications Interface, description of, 2-8
- Communications Manager, 9-6
- communications system services, 9-8
  - optimizing performance, 15-10
- compact disc (CD), 12-6
- Config.sys*, 5-2, 16-1. *See also* operating system configuration file.
- configuration
  - Cluster File Access, 10-2
  - definition of, 1-2, 1-3
  - files
    - Command Access Service, 6-15
    - Config.sys*, 14-3, 16-1

- FormatDiskConfig.sys*, 11-13, 11-33  
*SrpConfig.sys*, 14-3, 16-1  
 tape, 12-4  
 user, 7-1  
*UserCmdsConfig.sys*, 6-15  
**Configure Disk** command, 6-2, 6-3, 11-28  
**Configure Sequential Access Device** command, 12-4  
 configuring  
   operating system, 16-1. *See also* operating system configuration file.  
   tape drives, 12-4  
 Context Manager, memory partitions, 15-1  
 corrupted volume, 11-33  
   backing up, 13-14  
   identifying, 13-13  
   recovering, 13-12  
   reinitializing, 11-33  
   restoring data, 13-15  
   volume home block (VHB), 13-14  
*:cParExitRunFile:*, 16-16  
*:cParSpecHeap:*, 16-16  
*:cParSysCommonHeap:*, 16-16  
 CPU. *See* processor.  
 crash dumps, 19-25  
   extended crash dump file, 19-28  
   ISA-bus and other  
     single-processor workstations, 19-27  
   multiprocessor (EISA-bus) workstations, 19-28  
   SRP, 19-29  
   suppressing, 19-29  
   X-Bus workstations, 19-26  
   X-Bus+ workstations, 19-27  
   crash. *See* system crash.  
*CrashDump.sys*, 19-26  
*:CrashDumpPath:*, 16-17, 19-30  
**Create Directory** command, 6-7  
**Create File** command, 19-29  
**Create Message File** command, 18-3  
*:CreateDirectoryProtection:*, 16-17  
 CTOS  
   definition of, 1-2  
   Development Utilities, 17-3  
   I/O Processor Card (IOP). *See* I/O Processor Card.  
   SRP, 3-5  
   System Build Utilities, 17-2  
   VideoCard, 2-17  
   workstation, 3-2  
   XE, 3-5  
 customizing Standard Software, 18-1 to 18-4. *See also* message files.
- ## D
- Data Processor, description of, 2-9  
 DataComm Service, 9-2  
 Debugger, 4-15  
*:DebugPort:*, 16-19  
 default user file, 7-10  
 demand paging, 9-11, 15-4. *See also* paging service.  
 Development Utilities, 17-3  
 device  
   drivers, 9-4  
   names  
     SRP, 11-4  
     workstation, 11-2  
   passwords, 6-3, 13-14  
   templates, 11-6, 11-13, 11-23



## Index

---

- digital data storage (DDS) tape
    - approved, 12-2
    - configuring, 12-4
    - drives, 12-1
    - erasing, 12-5
  - directories
    - dollar-sign, 13-1, 15-8
    - operating system build, for, 17-4
    - passwords, 6-5
  - Disable Caching** command, 14-4
  - Disable Cluster** command, 13-2
  - disabling window interface, 18-7
  - disk
    - bad spots
      - file sample, 11-11
      - non-SCSI format, 11-10
      - report, 11-3
      - requirements, 11-6
      - SCSI format, 11-11
    - cleaning up, 13-1
    - compatibility with processors,
      - SRP, 11-4
    - device names
      - SRP, 11-4
      - workstation, 11-2
    - device templates, 11-6, 11-13,
      - 11-23, 11-24
    - EISA/ISA-bus workstations, 2-16,
      - 11-1
      - bootstrapping, A-1 to A-2, A-55
      - errors, A-55
    - error entry, **PLog**, 19-2
    - Format Disk** command, 11-6.
      - See also* the individual command name listing.
    - format templates, 11-13, 11-14
    - IDE, 11-2
    - non-SCSI workstation, 11-2
    - optimizing space, 11-33, 15-7
    - parameter templates, 11-13
    - partitioned, 11-3, 11-29
    - SCSI
      - SRP, 11-4
      - workstation, 11-2
    - server, accessing, 10-1
    - SMD, 11-4
    - SRP, 11-4
    - ST-506, 11-4
    - surface testing, 11-31, 13-15
    - type, 13-14
    - vendor code, 11-3
    - workstation, 11-1
  - disk-intensive applications, 15-9
  - Disk Partition Display Details**
    - command, 11-30
  - Disk Partition Set Active**
    - command, 11-29
  - :DiskAllocationLimit:*, 16-20
  - diskless workstation, 2-3
    - client, 3-5
    - file-system cache, 14-7
  - :DiskLogThreshold:*, 16-21
  - :DiskRetryCount:*, 16-21
  - distribution media, 8-1
  - dollar-sign directory, 13-1, 15-8
  - drives
    - IDE, 2-16, 11-2
    - SCSI, 2-16
  - dynamic queues, 15-6
- ## E
- E-box, 2-7
  - :ECom:*, 16-21
  - editing files. *See* Editor application.
  - Editor application
    - cursor, 4-7
    - deletion keys, 4-7
    - exiting, 4-6

- 
- files
    - closing, 4-7
    - copying, 4-8
    - deleting, 4-8
    - editing, 4-6
    - opening, 4-7
    - renaming, 4-8
    - saving, 4-7
  - replacing text, 4-8
  - searching, 4-8
  - starting, 4-6
  - user file options for, 7-22
  - Editor** command, 4-6
  - EISA**
    - cards, 2-17
    - configuration utility (SCU), 2-15
  - :EISA:*, 16-22
  - EISA/ISA-bus workstation**
    - Bootstrap menu, using, 5-6
    - bootstrapping, 5-1, 5-2, A-1 to A-2, A-55
  - BIOS**, 2-14
  - cards, 2-13, 2-16
  - CMOS**, 2-14
  - crash dump, collecting, 19-27, 19-28
  - CTOS VideoCard**, C-1
    - description of, 2-13
    - device names, 11-2
  - disks, 11-1
  - drives
    - IDE, 2-16
    - SCSI, 2-16
  - errors, A-55
  - hardware configuration, 2-15
  - I-bus converter, 2-19
  - I/O Processor Card**, 2-13
  - keyboard, 2-18
  - models, 2-14
  - mouse, 2-18
  - Setup program, 2-15
  - Standard VGA, configuring, C-1
    - See also* Standard VGA.
  - System Configuration Utility (SCU), 2-15
  - video, 2-17
  - electronic mail
    - isolating, 15-9
    - system services, 9-6
  - Enable Caching** command, 14-5
  - enclosures, SRP. *See* cabinets.
  - :EnterDebuggerOnFault:*, 16-23
  - environment, user file, 7-6
    - default, 7-6
    - multiple, 7-8
    - optional, 7-8
  - erasing tapes, 12-5
  - error codes
    - 31 (no such request), 8-14, 9-29
    - 33 (service not available), 9-29
    - 202 (directory full), 8-15
    - 203 (no such file), 9-29
    - 219 (access denied), 8-15
    - 220 (file in use), 8-15
    - 225 (no free file headers), 11-35
    - 230 (disk full), 8-15
    - 301 (I/O error), 11-21, 11-30, 13-8, 19-2
  - X-Bus boot ROM errors, A-1 to A-39
  - B24 boot ROM errors, A-40 to A-54
  - EISA/ISA workstation errors, A-55
  - SuperGen Series 5000 boot ROM errors, A-56 to A-73
  - hexadecimal
    - conversion chart, 19-39
    - LED patterns, 19-18

- SRP front panel and processor
    - LEDs, 19-24, 19-39, B-1
  - Ethernet Interface, description of, 2-8
  - Executive
    - color attributes, setting, 18-8
    - system services, installing with, 9-15, 9-19, 9-22
    - user file options for, 7-16
  - expansion cards, 2-5
  - :ExtCrashVDMFile:*, 19-26
  - :ExtCrashDumpFile:*, 19-26
  - extended crash dump file, 19-28
- F**
- FDISK** command (DOS), 11-30
  - file
    - access modes, 6-5
    - caching, 14-1. *See also* file-system cache.
    - passwords, 6-9
    - protection levels, 6-6
  - File Processor, description of, 2-9
  - file-system cache, 14-1
    - agent cache, 14-6, 14-7
    - Cluster IOP, 14-7
    - configuring, 14-3
    - disabling files, 14-4
    - diskless workstation, 14-7
    - enabling files, 14-5
    - file attributes, setting, 14-4
    - local, 14-6, 14-8
    - locking files into, 14-5
    - parameters, 14-3, 14-8
    - RAM disk emulation, 14-5
    - real mode processors, 14-8
    - remote, 14-7
    - server, files from, 14-6
    - unlocking files, 14-6
    - write-behind, 14-3
    - write-through, 14-3
  - file-system prefix files, assembling and linking, 17-7
  - :FileCacheDefault:*, 14-3, 16-24
  - :FileCacheDefaultEnable:*, 14-4, 16-25
  - :FileCacheService:*, 14-3, 16-26
  - :FileStructureVerify:*, 16-27
  - floppy disks
    - bootable, 19-32
    - initializing, 11-7, 11-15
  - Font Service, 9-4
  - :ForceBlackBackground:*, 16-27
  - Format Disk** command, 11-6
    - configuration file format, 11-13
    - device templates, 11-6, 11-24, 11-13, 11-23
    - format templates, 11-13
    - parameter fields, 11-7
    - parameter templates, 11-13
  - format templates, 11-13, 11-14
  - formatting disks. *See* initializing disks.
- G**
- General Processor, description of, 2-8
  - Generic Print System, system services, 9-4
  - global policy, 15-5
  - global thrashing, 15-5
- H**
- half-inch tape
    - configuring, 12-4
    - drives, 12-1
    - erasing, 12-5

- write-enabling, 12-5
- hardware
  - configuration programs,
    - EISA/ISA-bus workstations, 2-15
  - IDs, 5-7
  - shared resource processor, 2-6
- hexadecimal errors codes
  - conversion chart, 19-39
  - LED patterns, 19-18
  - HwNNN>SignOn.txt*, 5-7, 18-8
  - HwNNN>SysInit.jcl*, 5-7
  - HwNNN>XVTSignOn.txt*, 18-8
- I
- I-bus converter, 2-19
- I/O errors, 11-30, 13-13
- I/O Processor Card
  - bootstrapping, 5-2
  - definition of, 1-2
  - file system cache, on, 14-7
- IDE drive, 11-2
- Indexed Sequential Access Method (ISAM)
  - isolating, 15-9
  - system service, 9-8
- initializing disks
  - bad spots
    - file format, 11-10
    - file sample, 11-11
    - requirements, 11-6
  - corrupted volumes, 11-33
  - device templates, 11-6, 11-13, 11-23, 11-24
  - Format Disk** command, 11-6
  - format templates, 11-13, 11-14
  - I/O errors, 11-32
  - parameter templates, 11-13
  - unformatted, 11-5
  - valid volumes, reinitializing, 11-12
- input/output (I/O) errors, 11-30, 13-13
  - Bad Sector** command, 11-31
  - PLog** command, 11-31, 19-2
  - reinitializing, 11-32
  - surface tests, 11-31
- Install** command, 8-4
- Install CDROM Service**
  - command, 12-7
- Install CFA File Filter** command, 10-4
- Install CFA Server Service**
  - command, 10-3
- Install CFA Workstation Agent**
  - command, 10-3
- Install Command Access Service** command, 6-14
- Install.log*, 8-11
- Install MCR Service** command, 7-13
- Install Sequential Access Service** command, 12-3
- Installation Manager** command
  - failures, recovering from, 8-12
  - Install Media menu, 8-5
  - Installation Parameters* form, 8-6
  - log file, 8-11
  - parameters, 8-7
  - public installation, 8-11
  - QIC tape installation, 8-10
  - removing an application, 8-12
  - restarting, 8-13
  - server, installation from, 8-10
  - Software Operations menu, 8-4
  - user file options for, 7-17
  - using, 8-4
- installing software. *See* software installation.

## Index

---

Intel processing units, 2-3  
interprocessor data transfers, 15-10  
IOP. *See* I/O Processor Card.  
ISA cards, 2-16, 2-17  
*:ISOVideoCompliant:*, 16-28

## J

Job Control Language (JCL)  
  multiprocessor systems, 9-22  
  RunNoWait, 9-27  
  statements, 9-18  
  syntax, 9-24 to 9-26

## K

keyboard, 2-6  
  Cluster View, 4-17  
  EISA/ISA-bus workstations, 2-18  
keyswitch, SRP, 5-11  
  files  
    *SrpConfig.k.sys*, 16-9  
    *SysInit.k.jcl*, 9-24  
  positions, 9-24  
keywords  
  operating system configuration  
    file, 16-10  
  user file, 7-8, 7-14

## L

LEDs, 19-20, A-5 to A-6  
LFS (local file system), 3-4  
limiting access. *See also* restricting  
  system access.  
  commands, to, 6-14  
  directories, to, 6-9  
  system, 7-10

**Link CTOS** command, 17-10  
**Link CTOS I** command, 17-10  
**Link CTOS II** command, 17-10  
**Link CTOS III** command, 17-10  
**Link CTOS VM** command, 17-10  
linking operating systems, 17-7  
**List Message File** command, 18-2  
*.lst* files, 13-2  
loadable requests, 8-13  
*:LoadableRequestFile:*, 16-31  
*:LoadDebugger:*, 16-31  
local caching, 14-6, 14-8  
local file system (LFS), 3-4  
**Lock In Cache** command, 14-6  
Login Service, 4-17  
*Logins.sys*, 6-16  
*LogUnknownEntries:*, 16-31

## M

Mail Service, 9-6  
*:MassStorage:*, 15-12, 16-32  
master processor, 2-9  
Math Service, 9-2  
*:MaxXBlocksOut:*, 16-38  
MCR Service, 9-2  
memory  
  blocks, 15-10  
  code sharing, 15-4  
  disk, 15-12  
  expansion boards, 2-11  
  management  
    partitioned, 9-11  
    virtual, 9-11  
  partitions, 15-1  
*:MemoryScrub:*, 16-37  
**Merge Binary Message Files**  
  command, 18-4

**Merge Message Files** command, 18-4

message files, 18-1

- binary, 18-1, 18-3
- customizing, 18-2
- merging, 18-4
- SignOnMsg.bin*, 18-7
- Standard Software, 18-1
- text files, 18-2

Mini-Customizer, 17-1, 17-2

**Mini Customizer** command, 17-2

Modem Service, 9-6

modify access, 6-5

modules

- disks, 11-1
- workstation, 2-5

monitors, 2-5

mouse

- EISA/ISA-bus workstations, 2-18
- System Manager, using with, 4-3
- user file options for, 7-16

Mouse Service, 9-2

**Multiprocessor Statistics**

- command, 15-12

multiprocessor workstation

- booting, 5-2
- Command Access Service, installing, 6-14
- crash dump, collecting, 19-28
- definition of, 1-2
- Partition Status** command, 9-13
- system initialization file, 9-20
- System Manager**, using on, 4-2
- system services, installing, 9-19

## N

network system services, 9-8

non-SCSI tape drives, 12-1, 12-2

notebook workstation, 1-2, 5-6

*:nRepollActive:*, 16-40

*:nRkvsUsers:*, 16-40

## O

office automation applications, 8-2

OFIS Mail. *See* electronic mail.

*-old* files, 13-2

operating system

- building a customized, 17-3
- configuration file
  - boot section, 16-6
  - Bootstrap** command, specifying with, 16-9
  - defaults, 16-7
  - editing, 16-8
  - keyswtich files, 16-9
  - multiprocessor workstation, 16-4
  - processor section, 16-6
  - single-processor workstation, 16-3
  - special purpose, 16-9
  - SRP, 16-5
  - syntax, 16-2
- definition of, 1-2
- prefix files, assembling and linking, 17-9
- types, 3-3
- SRP, 3-5



## Index

---

- workstation, 3-2
  - `:OSNameClstr:`, 16-41
  - `:OSNameClstrLfs:`, 16-41
  - `:OSNameSrvr:`, 16-41
  - `:OSNameStnd:`, 16-41
- P**
- page faults, 15-4
  - paging service
    - global policy, 15-5
    - page faults, 15-4
  - parameter templates, 11-13
  - Partition Status** command, 19-5
    - code sharing, 15-4
    - display, 9-12
    - global thrashing, 15-5
    - multiprocessor workstation, 9-13
    - single-processor system, 9-13
  - partitioned disks, 11-3
    - bootable partition, 11-29
    - formatting, 11-29
  - partitions, memory, 15-1, 9-11
  - passwords
    - changing, volume, 6-4
    - device, 13-14
    - directory, 6-5
    - file, 6-9
    - protection levels, 6-6
    - SignOn, 6-12
    - <Sys> directory, 6-9
    - system volume, 6-2
    - user files, assigning to 7-11
    - volume, 6-1
  - performance optimization
    - Context Manager, 15-1
    - disk-intensive applications, 15-9
    - disk space utilization, 15-7
    - file-system cache, 14-1
    - interprocessor data transfers, 15-10
    - memory blocks, 15-10
    - queues, 15-6
  - PLog** command, 11-31, 13-13, 19-1
    - examples, 19-2
  - prebuilt operating systems
    - SRP, 3-5
    - workstation, 3-3
  - primary cabinet, disks in, 11-5
  - Print Service, 9-4
  - processor
    - crash entries, **PLog**, 19-3
    - identifiers, 2-10
    - LEDs, 19-16
    - protected-mode, 2-3, 2-8
    - real-mode, 2-3, 2-8
    - SRP, 2-8
  - `:Processor:`, 16-6, 16-43
  - protected-mode operating system
    - SRP, 3-5
    - workstation, 2-3, 3-2
  - protection levels, 6-6
- Q**
- quarter-inch cartridge (QIC) tape
    - approved, 12-2
    - configuring, 12-4
    - drives, 12-1
    - erasing, 12-5
    - retensioning, 12-5
  - queue
    - dynamic, 15-6
    - static, 15-6, 15-7
  - Queue.index*, 15-7
  - Queue Manager, 9-2, 9-4

**R**

RAM disk, 14-5, 15-12  
 read access, 6-5  
**Read Hardware ID** command, 5-7  
 real-mode operating systems  
   SRP, 3-5  
   workstation, 3-2  
 real-mode processor, 2-3  
 release documentation, 8-1  
 remote caching, 14-7  
 Remote Keyboard Video Service (RKVS), 4-11, 9-2  
 Remote User Manager (RUM), 4-11, 9-3, 9-27  
*:RemoteCacheClient:*, 14-8, 16-43  
*:RemoteCachePool:*, 14-8, 16-44  
*:RemoteCacheService:*, 16-45  
 request files, 8-13  
*Request.sys*, 8-14  
*:RequestTracker:*, 16-46  
 resource files, C-1  
**Restore Archive** command  
   complete volume backup, 13-9  
   corrupted volume, 13-12  
   parameter fields, 13-10  
 restoring backups  
   complete, 13-9  
   corrupted volume, 13-15  
   portions, 13-12  
 restricting system access, 6-11  
   videoless servers, 6-13  
 retensioning tapes, 12-5  
 RKVS, 4-11, 9-2  
*:RkvsFile:*, 16-46  
*rmSysGen.asm*, 17-3  
 RUM, 4-11, 9-3, 9-27  
 run files for system services, 9-15

**S**

saving files, 4-7  
*:sBroadcastHeap:*, 16-47  
*[Scr]* volume, 15-8  
 scratch volume, 15-8  
*:ScratchVolumeName:*, 15-8, 15-13  
 Screen Print Service, 9-3  
**Screen Setup** command, 18-8  
 SCSI  
   device names, 12-7  
   tape drives, 12-1  
 SCSI Interface, 2-8  
*:SCSIManagerName:*, 16-48  
 SCU, 2-15  
 search and replace. *See* Editor application.  
 Sequential Access Service, 9-3  
   installing, 12-3  
   parameters, 12-4  
*:SequentialStorage:*, 16-49  
 Series 1000, 1-2. *See also* notebook workstation.  
 Series 2000, 3000, 4000, 6000, and 7000, 2-3, 2-14  
 Series 5000, 2-3  
 server  
   bootstrapping from, 19-31  
   definition of, 1-2, 2-1  
   disks, accessing, 10-1  
   high-end EISA, 2-14  
   videoless workstation, 6-13  
   workstation operating systems, 3-4  
   Server Agent, 15-9  
**Set Directory Protection**  
   command, 6-8, 6-9

- Set Protection** command, 6-10, 6-11, 6-12
- Set Time** command, with Command Access Service, 6-14
- Setup program, 2-15
- shared resource processor
  - 186-board LEDs, B-2, B-6
  - 386-board LEDs, B-9
  - bootstrapping, 5-8
    - from tape, 19-37
  - cabinets, 2-6
  - Cluster View, using on, 4-10
  - Command Access Service, installing, 6-14
  - configuring operating system, 16-1
  - crash dumps, collecting, 19-29
  - decimal LEDs, 19-20
  - definition of, 2-6
  - disks, 11-4
  - error codes
    - boot media, B-4
    - disk hardware, B-4, B-6
    - expansion enclosure, B-6
    - processor, B-5
    - software, B-5
  - file-system cache, 14-3
  - front panel codes, B-3
  - General Processor LED display, B-2
  - hardware, 2-6
  - hexadecimal LED patterns, 19-18
  - installing system services, 9-19
  - keyswitch, 5-11, 9-24
  - master processor, 2-9
  - memory expansion boards, 2-11
  - operating systems, 3-5
    - configuration file, 16-1
  - optimizing memory, 15-9
  - primary cabinet, disks in, 11-5
  - processor identifiers, 2-10
  - protected-mode processors, 2-8
  - real-mode processors, 2-9
  - slot numbers, 19-4
  - System Build Utilities, 17-2
  - system crash status words, B-12
  - system initialization, 9-20
    - sequence, 5-10
  - troubleshooting
    - bootstrap problems, 19-22
    - front panel, 19-16
    - hardware installation, 19-15
    - minimum configurations, 19-23
    - processor crash, 19-15
    - processor LEDs, 19-16
  - tape drives, 12-1
- signing on
  - automatically, 7-12
  - default environment, 7-6
  - MCR Service, with, 7-13
  - optional environment, 7-8
  - without a user name, 7-10
- SignOn**
  - chain file, 7-10
  - color attributes, setting, 18-9
  - customizing, 18-5
    - conventional, 18-5
    - heading, 18-7
    - text file, 18-7
    - window interface, 18-6, 18-7
  - environments, 7-6
  - exit file, 7-10
  - password, 6-12
  - SignOn.txt*, 18-7
  - SignOnMsg.bin*, 18-7
  - user file options for, 7-14
  - XVTSignOn.txt*, 18-7
  - SignOn.txt*, 18-7
  - SignOnMsg.bin*, 18-7

- single-processor workstation,
  - conserving memory on, 15-9
- slot numbers, SRP, 19-4
- SMD disks, SRP, 11-4
- SNA Network Gateway, 9-8
- software installation
  - common problems, 8-14
  - compact disc (CD), installation
    - from, 8-10
  - errors, 8-15
  - failures, recovering from, 8-12
  - Installation Manager**
    - command, 8-3
    - planning, 8-1
    - public, 8-11
    - QIC tape, installation from, 8-10
    - restarting, 8-13
    - server, installation from, 8-10
  - Software Installation** command, 8-4
  - software package, 8-1
  - source code, for operating system
    - build, 17-3
  - Spooler (pre-GPS), 9-3
  - SrpConfig.sys*, 5-9, 5-11, 16-1
  - :SrvrAgentFile:*, 15-9, 16-52
  - Standard Software, 3-1
    - customizing, 18-1
    - definition of, 1-2
    - system services, 9-2
    - template files, 18-4
  - Standard VGA, 2-18
    - attributes
      - mapping theory, C-5
      - remapping, C-2, C-5
    - bold-for-underline substitution, C-3
    - color mapping, C-2
    - configuring, C-1
    - current attributes, displaying, C-4
    - extensive modifications, C-5
    - struck-through mapping, C-3
    - underline-for-bold substitution, C-4
    - VGA Character Attribute Mapper** command, C-5
  - static partitions, 15-3
  - static queues, 15-6, 15-7
  - Statistics Service, 9-3
  - status codes
    - CTOS, initialization, A-3 to A-4.
      - See also* error codes.
    - SRP front panel and processor LEDs, B-1
  - Storage Processor, description of, 2-9
  - ST-506 disks, SRP, 11-4
  - SuperGen, 2-3, 2-14
  - :SuppressAutoDump:*, 16-53, 19-29
  - surface testing disks, 11-31
  - :SwapFile:*, 16-55
  - :SwapFileSize:*, 16-56
  - :SwapFileSizeMax:*, 16-56
  - [Sys], 5-1
  - SysGen, 17-1. *See also* building an operating system.
  - SysGen.asm*, 17-4
  - SysImage.sys*, 5-1
  - SysInit.jcl*, , 5-2, 5-10, 9-16
  - SysInit.k.jcl*, 9-24
  - System Build Utilities
    - CTOS/XE, 17-2
    - Mini-Customizer, 17-1, 17-2
    - workstation, 17-2
  - System Configuration Utility (SCU), 2-15
  - system crash
    - intermittent, 19-25
    - PLog** error entry, 19-3
    - SRP, 19-15

## Index

---

- system image, 5-1
- system initialization
  - error entry, **PLog**, 19-3
  - file
    - creating, 9-17
    - multiprocessor workstation,
      - sample of, 9-21
    - single-processor workstation,
      - sample of, 9-17
    - SRP, sample of, 9-23
    - syntax, 9-24
  - SRP, 5-10, 5-11
  - workstation, 5-2
- System Manager
  - function keys, 4-4
  - keyboard, 4-3
  - mouse, 4-3
  - multiprocessor workstation, using
    - on, 4-2
  - SRP
    - Cluster View, starting with, 4-4
    - display, 4-5
    - using on, 4-4
  - workstation display, 4-2
- System Manager** command, 4-2
- system optimization. *See*
  - performance optimization.
- system security, 6-1
- system services
  - CD-ROM Service, 9-2
  - Cluster File Access, 9-2, 10-1
  - Command Access Service, 9-2
  - commands, 9-15
  - Communications Manager, 9-6
  - DataComm Service, 9-2
  - definition of, 9-1
  - device drivers, 9-4
  - electronic mail, 9-6
  - Font Service, 9-4
  - Generic Print System, 9-4
  - installing
    - SRP, 9-22
    - multiprocessor workstation, 9-19
    - single-processor workstation,
      - 9-14
  - Job Control Language
    - statements, 9-18, 9-22
    - syntax, 9-24
  - Math Service, 9-2
  - MCR Service, 9-2
  - memory requirements, 9-11
  - Modem Service, 9-6
  - Mouse Service, 9-2
  - network, 9-8
  - Print Service, 9-4
  - Queue Manager, 9-2, 9-4
  - Remote Keyboard Video Service (RKVS), 9-2
  - Remote User Manager (RUM),
    - 9-3, 9-27
  - run files, 9-15
  - Screen Print Service, 9-3
  - Sequential Access Service, 9-3,
    - 12-3
  - Spooler (pre-GPS), 9-3
  - Standard Software, 9-2
  - Statistics Service, 9-3
  - system initialization file
    - creating, 9-17
    - multiprocessor workstation,
      - sample of, 9-21
    - SRP, sample of, 9-23
    - workstation, sample of, 9-17
  - Voice Service, 9-3
  - where to install, 9-9
  - XBIF Service, 9-3, 12-2
  - XC-002 Service, 9-3
  - XVT/CH, 18-6

system software  
 operating system  
 SRP, 3-5  
 workstation, 3-2  
 Standard Software, 3-1  
 Video Access Method (VAM), 3-1  
 system volume, 5-1

## T

tape  
 backups. *See* backups.  
 bootable, SRP, 19-37  
 drives  
 configuring, 12-4  
 digital data storage (DDS), 12-1  
 half-inch, 12-1  
 non-SCSI, 12-1, 12-2  
 quarter-inch cartridge (QIC),  
 12-1  
 SCSI, 12-1  
**Tape Erase** command, 12-5  
**Tape Retension** command, 12-5  
 template files, 18-4  
 Terminal Processor, description of,  
 2-9  
 testing customized operating  
 systems, 17-10  
 troubleshooting  
 bootstrapping  
 floppy diskette, from, 19-32  
 server, from, 19-31  
 tape, from, 19-37  
 crash dumps, 19-25  
 diagnosing problems  
**Cluster Status** command, 19-4  
**Partition Status** command,  
 19-5  
**PLog** command, 19-1  
 disk errors, 13-14, 19-2

software installation, 8-14  
 SRP  
 bootstrap, 19-37  
 front panel, 19-16, B-1  
 hardware installation, 19-15  
 LEDs, 19-16  
 minimum configurations, 19-23  
 processor crash, 19-15, B-1  
 SysGen errors  
 assembly, 17-13  
 bootstrap, 17-14  
 link, 17-13  
 system crash, 19-3  
 system services, 9-29  
 workstation  
 applications, 19-13, 19-14  
 bootstrapping, 19-7  
 clustering, 19-12  
 keyboard, 19-10  
 monitor, 19-11  
 power, 19-6  
 problems, 19-5  
 X-Bus module, 19-8  
 .ts files, 13-2  
 .txt files, 18-2

## U

unformatted disks, initializing,  
 11-5  
**Unlock Cache** command, 14-6  
 .user, 7-10  
 user configuration files  
 automatic SignOn, 7-12  
 creating, 7-1  
 default, removing, 7-11  
 editing, 7-3  
 Editor, 7-16  
 environments, 7-6  
 file specifications for, 7-4



- format of, 7-4
- keywords, 7-4, 7-14
- limiting system access with, 7-10
- options, 7-14
  - adding, 7-5
  - changing, 7-3
  - Cluster View, 7-20
  - Editor, 7-22
  - Executive, 7-16
  - Installation Manager, 7-17
  - mouse, 7-16
  - SignOn, 7-14
- passwords, assigning to, 7-11
- sample of, 7-4, 7-7
- SignOn
  - chain file, 7-10
  - exit file, 7-10
- User File Editor** command
  - Functions menu, 7-2
  - options entry form, sample of, 7-2
  - passwords, assigning with, 7-11
  - user file, creating with, 7-1
  - working environment, creating with 7-6
- user file environments, 7-6
  - default, 7-6
  - multiple, 7-8
  - optional, 7-8
- user files. *See* user configuration files.
- user names, 7-2. *See also* user configuration files.
  - eliminating, 6-12
  - prepackaged, 6-12
- UserCmdsConfig.sys*, 6-15

## V

- valid volume, 11-1
  - reinitializing, 11-12
- variable partitions, 15-3
- Vdm\_Ch.run*, 6-13
- VdmAttributes.res*, C-1
- :VDMFile:*, 16-57
- vendor code, disk modules, 11-3
- Version** command, 15-2
- VGA Add Color Assignment**
  - command, C-2
- VGA Add Struck-Through**
  - command, C-3
- VGA Bold Replaces Underline**
  - command, C-4
- VGA Character Attribute Mapper** command
- VGA Display Mapping** command, C-4
- VGA Underline Replaces Bold**
  - command, C-4
- video
  - Cluster View, 4-17
  - EISA/ISA-bus workstations, 2-17
  - Standard VGA, 2-18
- Video Access Method (VAM), 3-1
- Video** command, 18-8
- videoless server, 6-13
- virtual memory, 9-11
  - demand paging, 9-11, 15-4
  - operating system, 3-2
  - optimizing, 15-4
  - working set, 15-5
- Voice Service, 9-3

- volume
    - compact disc, 12-6
    - corrupted, 11-33
    - home block (VHB), 13-14
    - passwords, 6-1
    - scratch, 15-8
    - valid, 11-1
  - Volume Archive** command
    - full volume backup, 13-2
    - incremental backup, 13-6
    - parameter fields, 13-3
  - Volume Status** command, 11-33
- W**
- W-blocks, 15-11, 15-12
  - :*WakeUpInterval*:, 16-58
  - :*WatchDogStatus*:, 16-58
  - :*WBlocks*:, 15-12, 16-59
  - :*WBlocksIn*:, 16-59
  - :*WBlocksOut*:, 16-59
  - window interface, SignOn, 18-6
  - working set, 15-54
  - workstation
    - B26, B28, B38, and B39, 2-2
    - bootstrapping, 5-1
      - floppy diskette, from, 19-32
      - server, from, 19-31
    - cartridges, 2-5
    - Cluster View, using on, 4-10
    - configuring operating system, 16-1
    - conserving memory, single
      - processor, on, 15-9
    - crash dump, collecting, 19-26
    - diskless, 2-3
    - disks, 11-1
    - EISA/ISA-bus, 2-13. *See also* EISA/ISA-bus workstation.
    - expansion cards, 2-5
    - file-system cache, 14-3
    - hardware, 2-2
    - system services
      - multiprocessor, installing, 9-19
      - single processor, installing, 9-14
    - keyboards, 2-6
    - modules, 2-5
    - monitors, 2-5
    - notebook, 1-2, 5-6
    - operating systems, 3-2
    - processors, 2-3
    - Series 1000, 1-2. *See also* notebook workstation.
    - Series 2000, 3000, 4000, 6000, and 7000, 2-3, 2-14
    - Series 5000, 2-3
    - setup, 1-4
    - SuperGen, 2-3
    - System Build Utilities, 17-2
    - system initialization, 5-2, 9-16
    - tape drives, 12-1
    - troubleshooting
      - applications, 19-13, 19-14
      - bootstrapping, 19-7
      - clustering, 19-12
      - keyboard, 19-10
      - monitor, 19-11
      - power, 19-6
      - X-Bus module, 19-8
    - type number, 5-3
    - X-Bus+, 2-3
    - X-Bus, 2-2
  - workstation-type number, 5-3
  - write-behind cache, 14-3
  - write-enabling half-inch tapes, 12-5
  - Write Hardware ID** command, 5-7
  - write-through cache, 14-3

## Index

---

*Ws>SignOn.txt*, 18-8  
*Ws>XVTSignOn.txt*, 18-8  
*WsNNN>Config.sys*, 5-3  
*WsNNN>SignOn.txt*, 5-3, 18-8  
*WsNNN>SysImage.sys*, 5-3  
*WsNNN>SysInit.jcl*, 5-3, 9-17  
*WsNNN>XVTSignOn.txt*, 18-8

## X

X.25 Network Gateway, 9-8  
X-blocks, 15-11  
X-box, 2-7  
X-Bus workstation, 2-2  
    boot ROM errors, A-1 to A-39  
    crash dump, collecting, 19-26  
X-Bus+ workstation, 2-3  
    crash dump, collecting, 19-27  
XBIF Service, 9-3, 12-2  
*:XBlocks:*, 15-11, 16-60

*:XBlocksSmall:*, 15-11, 16-61  
XC-002 Service, 9-3  
**XESoftware Installation**  
    command, 8-4  
XVT/CH System Service, 18-6  
*XVTSignOn.txt*, 18-7

## Y

Y-blocks, 15-11, 15-12  
*:YBlocks:*, 15-12, 16-62  
*:YBlocksIn:*, 16-63  
*:YBlocksOut:*, 16-63

## Z

Z-blocks, 15-11, 15-12  
*:ZBlocks:*, 15-12, 16-63  
*:ZBlocksIn:*, 16-64  
*:ZBlocksOut:*, 16-64



43574599-300