

# Multics

## Operator's Guide

MULTICS  
OPERATOR'S GUIDE  
ADDENDUM B

SUBJECT

Additions and Changes to the Manual

SPECIAL INSTRUCTIONS

This is the second addendum to GB61, Revision 1, dated May 1985. Throughout Section 11, change bars in the margins indicate technical additions and changes; asterisks denote deletions. Insert the attached pages into the manual according to the collating instructions on the back of this cover.

Refer to the Preface for "Significant Changes."

**Note:** Insert this cover behind the manual cover to indicate the updating of this document with Addendum B.

SOFTWARE SUPPORTED

Multics Software Release 12.1

ORDER NUMBER

GB61-01B

December 1987

51322  
1088  
Printed in U.S.A.

**Honeywell Bull**

## **COLLATING INSTRUCTIONS**

To update this manual, remove old pages and insert new pages as follows:

### **Remove**

Front cover, blank  
Title page, Preface  
iii through xiii  
Section 11  
Index  
Blank, rear cover

### **Insert**

Front cover, blank  
Title page, Preface  
iii through xii  
Section 11  
Index  
Blank, rear cover

**Honeywell Bull disclaims the implied warranties of merchantability and fitness for a particular purpose and makes no express warranties except as may be stated in its written agreement with and for its customer. In no event is Honeywell Bull liable to anyone for any indirect, special or consequential damages.**

**The information and specifications in this document are subject to change without notice. Consult your Honeywell Bull Marketing Representative for product or service availability.**

# CONTENTS

PART I	Introduction	
Section 1	Introduction . . . . .	1-1
	How to Use This Manual . . . . .	1-1
	Manual Conventions . . . . .	1-2
	Other Manuals of Interest . . . . .	1-2
	People to Know About . . . . .	1-3
	Honeywell Procedures vs Those at Your Site . . . . .	1-4
PART II	System Description	
Section 2	Hardware Overview . . . . .	2-1
	Major Modules . . . . .	2-1
	Memory and System Controller Unit (SCU) . . . . .	2-1
	Central Processing Unit (CPU) . . . . .	2-1
	Input/Output Multiplexer (IOM) . . . . .	2-1
	Information Multiplexer Unit (IMU) . . . . .	2-2
	Front-End Network Processor (FNP) . . . . .	2-3
	Microprogrammed Peripheral Controller (MPC) . . . . .	2-3
	Peripherals . . . . .	2-5
	Terminals . . . . .	2-5
	Storage Devices . . . . .	2-5
	Disks . . . . .	2-5
	Mass Storage Processor (MSP) . . . . .	2-5.1
	Tapes . . . . .	2-5.1
	Magnetic Tape Processor (MTP) . . . . .	2-5.1
	Unit Record Devices . . . . .	2-5.2
	Card Punches . . . . .	2-6
	Card Readers . . . . .	2-6
	Unit Record Processor (URP) . . . . .	2-6
	Panels . . . . .	2-6
	DPS 8 vs Level 68 . . . . .	2-6
Section 3	Software Overview . . . . .	3-1
	Storage System . . . . .	3-1
	Disk Volumes . . . . .	3-1
	Operating System . . . . .	3-5
	Bootling . . . . .	3-5
	Bootload Command Environment (BCE) . . . . .	3-6
	Multics . . . . .	3-7
	System Startup . . . . .	3-8
	System Shutdown . . . . .	3-8
	Crash . . . . .	3-8
	Emergency Shutdown . . . . .	3-8
	Configuration . . . . .	3-9
	System Software . . . . .	3-9

	Hardcore Supervisor . . . . .	3-10
	Online Software . . . . .	3-10
	Processes . . . . .	3-10
	Initializer Process . . . . .	3-10
	Rings 1 and 4 . . . . .	3-11
	Message Coordinator . . . . .	3-11
	Resource Control Package (RCP) . . . . .	3-11
	Answering Service . . . . .	3-12
	exec_coms . . . . .	3-12
	system_start_up.ec . . . . .	3-12
	admin.ec . . . . .	3-12
	Daemons . . . . .	3-12
	Backup . . . . .	3-13
	Backup Functions . . . . .	3-13
	Backup Systems . . . . .	3-14
	Salvagers . . . . .	3-14
	Firmware . . . . .	3-15
	Calendar Clock . . . . .	3-15
	Load Control Groups . . . . .	3-15
	Logging . . . . .	3-15
	Comparison of Multics and GCOS Terms . . . . .	3-16
Section 4	The Operator Interface . . . . .	4-1
	Your Responsibilities . . . . .	4-1
	Your User_id . . . . .	4-2
	Commands and Arguments . . . . .	4-2
	Operator Authentication . . . . .	4-3
	Signing On . . . . .	4-3
	Signing Off . . . . .	4-4
	Getting Help . . . . .	4-4
	Using Special Characters . . . . .	4-6
PART III	Communicating with the System	
Section 5	Using The Bootload Console . . . . .	5-1
	Typing Input . . . . .	5-1
	Issuing Commands . . . . .	5-2
	Correcting Errors . . . . .	5-2
	A Note about Numbers . . . . .	5-3
	Entering Commands . . . . .	5-3
	Interrupting the System . . . . .	5-4
	Interrupting the System From Multics . . . . .	5-5
	Commands You Can Issue at the Bootload Console . . . . .	5-5
	Messages on the Bootload Console . . . . .	5-5
	The Audible Alarm . . . . .	5-5
	Rerouting Bootload Console Activity to a Regular User Terminal . . . . .	5-5
	Communicating with the MCA(s) . . . . .	5-6
	Problems . . . . .	5-7
	Summary of Bootload Console Keys . . . . .	5-8
Section 6	Using An Initializer Terminal . . . . .	6-1
	Issuing Commands . . . . .	6-1
	Correcting Errors . . . . .	6-1
	Entering Commands . . . . .	6-1

	Interrupting the System . . . . .	6-2
	Commands You Can Issue at the Initializer Terminal . . . . .	6-2
	Messages on the Initializer Terminal . . . . .	6-2
	Multiple Initializer Terminals . . . . .	6-2
	The Audible Alarm . . . . .	6-3
	Adding a Regular User Terminal to the Initializer Dynamically . . . . .	6-3
	Dropping A Terminal Dynamically . . . . .	6-4
	Rerouting Initializer Messages From One Terminal to Another . . . . .	6-4
	Putting Tabs Back On a TN300 . . . . .	6-5
Section 7	Dealing with System Messages . . . . .	7-1
	The Form of a System Message . . . . .	7-1
	BCE Messages . . . . .	7-2
	Syserr Messages . . . . .	7-2
	RCP Messages . . . . .	7-2
	Disk Error Messages . . . . .	7-2
	Salvager Messages . . . . .	7-3
	Message Coordinator Messages . . . . .	7-3
	Backup Daemon Messages . . . . .	7-3
	I/O Daemon Messages . . . . .	7-3
	Login and Logout Messages . . . . .	7-3
	Other Answering Service Messages . . . . .	7-4
	Initializer Command Responses . . . . .	7-4
	Error Message Documentation . . . . .	7-4
PART IV	Bringing the System Up and Shutting the System Down	
Section 8	Powering On the Machines . . . . .	8-1
	To Power on the Machines . . . . .	8-1
	Good Ideas . . . . .	8-2
Section 9	Checking Switch Settings . . . . .	9-1
	To Check the Switch Settings . . . . .	9-1
	To Set The Bootload IOM, SCU and CPU . . . . .	9-1
	Good Ideas . . . . .	9-2
Section 10	Bootloading BOS . . . . .	10-1
Section 11	Bootloading BCE . . . . .	11-1
	To Boot BCE From Scratch . . . . .	11-1
	Initializing BCE . . . . .	11-1
	Binary Representations . . . . .	11-2
	Loading Firmware into the Tape MPC Used to Boot BCE . . . . .	11-7
	Entering the Location of the Root Physical Volume (RPV) . . . . .	11-8
	Booting BCE . . . . .	11-8
	Setting the Calendar Clock . . . . .	11-9
	Editing the Clok Card . . . . .	11-9
	Setting the Time . . . . .	11-10
	Loading Firmware into the Disk MPCs . . . . .	11-11
	Loading Firmware into the Other MPCs . . . . .	11-11
	Resynchronizing the Paper in the Line Printers . . . . .	11-12
	Problems . . . . .	11-12
	Initializing BCE . . . . .	11-12
	Loading Firmware into the Tape MPC Used to Boot BCE . . . . .	11-13

	Booting BCE . . . . .	11-13
	Setting the Calendar Clock . . . . .	11-14
	Loading Firmware into the Disk and Other MPCs . . . . .	11-15
Section 12	Editing the Config Deck to Change Hardware States . . . . .	12-1
	Changing the State of a Processor . . . . .	12-1
	Changing the State of a System Controller . . . . .	12-2
	Changing the State of an IOM . . . . .	12-3
	Changing the State of an FNP . . . . .	12-4
	Changing the State of a System Console . . . . .	12-5
Section 13	Bootloading Multics . . . . .	13-1
	To Boot Multics From BCE . . . . .	13-1
	Problems . . . . .	13-2
Section 14	Logging in the Daemons -- I/O . . . . .	14-1
	To Start the I/O Daemons . . . . .	14-1
	To Log in and Start the I/O Daemon Coordinator . . . . .	14-3
	To Log in and Start a Printer I/O Daemon Driver . . . . .	14-3
	To Log in and Start a Punch I/O Daemon Driver . . . . .	14-5
	To Get Information About Devices and Request Types (and Device Classes) . . . . .	14-6
	To Find Out Which I/O Daemons Are Logged In . . . . .	14-8
	Problems . . . . .	14-8
Section 15	Logging In The Daemons -- Incremental Backup . . . . .	15-1
	To Start the Hierarchy Incremental Dumper . . . . .	15-1
	To Start the Volume Incremental Dumper . . . . .	15-2
	To Log in the Hierarchy Incremental Dumper . . . . .	15-2
	To Log in the Volume Incremental Dumper . . . . .	15-3
	To Find Out Which Hierarchy Backup Daemons Are Logged In . . . . .	15-3
	To Find Out Which Volume Backup Daemons Are Logged In . . . . .	15-3
Section 16	Bringing the System Up -- The Whole Procedure . . . . .	16-1
	Booting BCE and Multics . . . . .	16-1
Section 17	Shutting the System Down . . . . .	17-1
	To Do a Scheduled Shutdown . . . . .	17-1
	To Do a Nonscheduled Shutdown . . . . .	17-5
	To Shut Down Immediately . . . . .	17-5
	Problems . . . . .	17-5
Section 18	Powering Off the Machines . . . . .	18-1
	To Power Off the Machines . . . . .	18-1
PART V	Everyday Operations -- Storage Devices	
Section 19	Managing Tapes . . . . .	19-1
	To Mount a Tape . . . . .	19-1
	Beginning the Procedure . . . . .	19-2
	Authenticating a Tape . . . . .	19-3
	Denying a Tape . . . . .	19-3
	Finishing the Procedure . . . . .	19-4
	To Ready a Tape Drive . . . . .	19-4

	To Rewind a Tape . . . . .	19-4
	To Demount a Tape . . . . .	19-5
	To Preload a Tape . . . . .	19-5
	To Add a Tape Drive . . . . .	19-6
	To Delete A Tape Drive . . . . .	19-7
	To Get Information About Tape Mounts . . . . .	19-7
	Problems . . . . .	19-8
	Good Ideas . . . . .	19-8
Section 20	Managing User I/O Disks . . . . .	20-1
	To Mount a User I/O Disk Pack . . . . .	20-1
	To Ready a Disk Drive . . . . .	20-4
	To Convert a Disk Drive From User I/O Use to Storage System Use . . . . .	20-5
	To Convert a Disk Drive from Storage System Use to User I/O Use . . . . .	20-5
	To Add a Disk Drive . . . . .	20-5
	To Delete a Disk Drive . . . . .	20-6
	To Get Information About Disk Mounts . . . . .	20-6
Section 21	Managing Storage System Disks . . . . .	21-1
	To Mount a Public Logical Volume . . . . .	21-1
	Letting the System Decide Where to Mount Packs . . . . .	21-1
	Specifying Where Packs Are Mounted . . . . .	21-2
	To Mount a Private Logical Volume . . . . .	21-3
	To Ready a Disk Drive . . . . .	21-4
	To Cancel a Storage System Disk Mount . . . . .	21-4
	To Convert a Disk Drive from Storage System Use to User I/O Use . . . . .	21-5
	To Convert a Disk Drive from User I/O Use to Storage System Use . . . . .	21-5
	To Add a Disk Drive as a Storage System Drive . . . . .	21-6
	To Delete a Storage System Disk Drive . . . . .	21-6
	To Get Information About Disk Mounts . . . . .	21-8
	Problems . . . . .	21-8
PART VI	Everyday Operations -- Unit Record Devices	
Section 22	Managing I/O Daemons -- Printers . . . . .	22-1
	To Start Up the Printer . . . . .	22-2
	To Fix the Paper in the Middle of a Request . . . . .	22-2
	Beginning the Procedure . . . . .	22-3
	Fixing the Paper . . . . .	22-3
	Finishing the Procedure . . . . .	22-4
	To Reprint a Segment . . . . .	22-5
	To Change the Current Request Type Processing . . . . .	22-5
	To Find out the Status of the Printer . . . . .	22-6
	To Get Information About Printer Requests . . . . .	22-9
	To Kill a Printer Request . . . . .	22-9
	To Reprocess a Killed Request . . . . .	22-10
	To Defer a Printer request . . . . .	22-10
	To Reprocess a Deferred Request . . . . .	22-11
	To Reinitialize a Printer . . . . .	22-12
	To Log Out the Printer I/O Daemon Driver . . . . .	22-13
	Problems . . . . .	22-13
	Good Ideas . . . . .	22-13
Section 23	Managing I/O Daemons -- Card Punches and Readers . . . . .	23-1



	To Start Up the Card Punch . . . . .	23-1
	To Stop the Card Punch . . . . .	23-2
	To Restart the Card Punch . . . . .	23-2
	To Get Information About Punch Requests . . . . .	23-2
	To Read Cards . . . . .	23-3
	Checking Card Decks for Completeness . . . . .	23-3
	Preparing Card Decks for Reading . . . . .	23-4
	Starting Up the Daemon . . . . .	23-6
	To Stop the Card Reader . . . . .	23-8
	To Logout the Punch or Reader I/O Daemon Driver . . . . .	23-8
Section 24	Managing Remote Devices At The Central Site . . . . .	24-1
	To Log in and Start a Remote Printer/Punch . . . . .	24-2
	To Log in and Start a Remote Printer that Prints in Different Formats . . . . .	24-4
	To Change from One Print Format to Another . . . . .	24-7
	To Start Up a Type I Remote Station . . . . .	24-7
	To Start Up a Type II Remote Station . . . . .	24-9
	To Get Information About Communications Lines . . . . .	24-13
	Problems . . . . .	24-13
Section 25	Managing Remote Devices At a Remote Site . . . . .	25-1
	To Start Up a Remote Station . . . . .	25-1
PART VII	Everyday Operations -- Backup	
Section 26	Managing Backup Daemons -- Dumps . . . . .	26-1
	To Do an Incremental Hierarchy Dump . . . . .	26-2
	To Do an Incremental Volume Dump . . . . .	26-3
	To Do a Consolidated Hierarchy Dump . . . . .	26-4
	To Do a Consolidated Volume Dump . . . . .	26-5
	To Do a Complete Hierarchy Dump . . . . .	26-6
	To Do a Complete Volume Dump . . . . .	26-7
	To Logout a Dumper Daemon . . . . .	26-8
	Good Ideas . . . . .	26-9
Section 27	Managing Backup Daemons -- Retrievals . . . . .	27-1
	To Do a Volume Retrieval . . . . .	27-1
	To Do a Hierarchy Retrieval . . . . .	27-2
	To Logout a Retrieval Daemon . . . . .	27-5
	Problems . . . . .	27-5
	Good Ideas . . . . .	27-6
PART VIII	Everyday Operations -- Miscellaneous	
Section 28	Communicating with Users . . . . .	28-1
	To Send a Message to All Users at Login . . . . .	28-1
	To Send a Message to One User . . . . .	28-1
	To Send a Message to All Users . . . . .	28-2
	Problems . . . . .	28-2
Section 29	Managing User Logins, Logouts, and Channels . . . . .	29-1
	To Find Out How Many Users Are Logged In . . . . .	29-1
	To Find Out Which Users Are Logged In . . . . .	29-1

	To Set the Maximum Number of Users Who Can Log In . . . . .	29-1
	To Log a User Out . . . . .	29-2
	To Respond to a User Who's Been Logged Out . . . . .	29-3
	Problems . . . . .	29-4
Section 30	Managing Absentee Jobs . . . . .	30-1
	To Start the Absentee Facility . . . . .	30-1
	To Stop the Absentee Facility . . . . .	30-1
	To Set the Number of Absentee Slots . . . . .	30-2
	To Stop An Absentee Queue . . . . .	30-2
	To Restart an Absentee Queue . . . . .	30-3
	To Move an Absentee Job from One Queue to Another . . . . .	30-3
	To Set the Highest Numbered Queue Which Will Be Serviced . . . . .	30-4
	To Bump an Absentee Job . . . . .	30-4
	To Defer an Absentee Job . . . . .	30-5
	To Suspend an Absentee Job . . . . .	30-5
	To Release an Absentee Job . . . . .	30-6
	To Force a Job to Log In Next . . . . .	30-6
	To Get Information About Absentee Jobs . . . . .	30-7
Section 31	Setting Attended/Unattended Mode and Manual/Automatic Mode . . . . .	31-1
	To Set Unattended Mode . . . . .	31-1
	To Set Attended Mode . . . . .	31-2
	To Set Automatic Mode . . . . .	31-2
	To Set Manual Mode . . . . .	31-3
Section 32	Doing Dynamic Reconfiguration . . . . .	32-1
	To Add a Processor . . . . .	32-1
	To Delete a Processor . . . . .	32-2
	To Add Memory . . . . .	32-2
	Clearing the Store Units . . . . .	32-3
	Adding the Memory . . . . .	32-3
	To Delete Memory . . . . .	32-4
	To Add an IOM . . . . .	32-4
	To Delete an IOM . . . . .	32-5
	To Add an FNP . . . . .	32-5
	To Delete an FNP . . . . .	32-6
	To Add a Logical Channel . . . . .	32-6
	To Delete a Logical Channel . . . . .	32-7
	To Add an Alternate Bootload Console . . . . .	32-7
	To Change the Bootload Console . . . . .	32-8
	To Delete the Bootload Console . . . . .	32-9
Section 33	Leaving a Note in the System Log Book . . . . .	33-1
	Good Ideas . . . . .	33-1
PART IX	Special Operations	
Section 34	Using the DPU and the DMP/VIP . . . . .	34-1
	To Use the DPU . . . . .	34-1
	DPU Typing Conventions . . . . .	34-1
	DPU Naming Conventions . . . . .	34-1
	DPU Modes . . . . .	34-1
	Getting Into TM Mode From Ready Mode . . . . .	34-2

	Getting into VIP Mode from TM Mode . . . . .	34-3
	Getting into TM Mode from VIP Mode . . . . .	34-3
	Getting into Ready Mode from TM Mode . . . . .	34-3
	Finding Out Which Unit is Connected to the DPU . . . . .	34-3
	Changing the Unit Connected to the DPU . . . . .	34-4
	Getting Help With DPU Commands . . . . .	34-4
To Use the DMP/VIP . . . . .		34-4
	DMP/VIP Typing and Naming Conventions . . . . .	34-4
	DMP/VIP Modes . . . . .	34-5
	Getting Connected to the DMP/VIP . . . . .	34-5
To Display Configuration Panels . . . . .		34-5
DPU/DMP Command Summary . . . . .		34-6
	Ready Mode Commands (C? Prompt) . . . . .	34-6
	TM Mode Commands (OFL? Prompt) . . . . .	34-6
	VIP Mode Command (<UNIT> CMD Prompt) . . . . .	34-7
	CPU Commands . . . . .	34-7
Section 35	Failures that Crash the System . . . . .	35-1
	To Recognize a System Crash in Automatic Mode . . . . .	35-1
	To Respond to a System Crash in Automatic Mode . . . . .	35-1
	To Recognize a System Crash in Manual Mode . . . . .	35-2
	To Respond to a System Crash in Manual Mode . . . . .	35-2
	Problems . . . . .	35-2
Section 36	Failures That Do Not Crash the System . . . . .	36-1
	To Recognize a System Loop . . . . .	36-1
	To Recognize a System Hang . . . . .	36-1
	To Return to BCE . . . . .	36-1
	Executing Fault . . . . .	36-2
	Executing Switches on a Level 68 System . . . . .	36-3
	Placing Nonbootload CPUs in Step . . . . .	36-3
	Executing Switches . . . . .	36-3
	Results of Executing Switches . . . . .	36-3
	Taking Nonbootload CPUs Out of Step . . . . .	36-4
	Initializing Nonbootload CPUs . . . . .	36-4
	Executing Switches on a DPS 8 System . . . . .	36-4
	Placing Nonbootload CPUs in Step . . . . .	36-4
	Placing Bootload CPU in Step . . . . .	36-4
	Executing Switches . . . . .	36-5
	Results of Executing Switches . . . . .	36-5
	Taking Nonbootload CPUs Out of Step . . . . .	36-6
	Initializing Nonbootload CPUs . . . . .	36-6
	Executing Switches with a Different Bootload CPU (L68) . . . . .	36-6
	Placing Current Bootload CPU in Step . . . . .	36-6
	Setting Switches on SCUs . . . . .	36-6
	Taking New Bootload CPU Out of Step . . . . .	36-7
	Initializing New Bootload CPU . . . . .	36-7
	Executing Switches . . . . .	36-7
	Results of Executing Switches . . . . .	36-7
	Taking Nonbootload CPUs Out of Step . . . . .	36-8
	Initializing Nonbootload CPUs . . . . .	36-8
	Shutting Down BCE . . . . .	36-8
	Setting Switches on SCUs . . . . .	36-8
	Rebooting BCE . . . . .	36-8

	Executing Switches with a Different Bootload CPU (DPS 8) . . . . .	36-9
	Placing Current Bootload CPU in Step . . . . .	36-9
	Setting Switches on SCUs . . . . .	36-9
	Taking New Bootload CPU Out of Step . . . . .	36-9
	Initializing New Bootload CPU . . . . .	36-9
	Placing New Bootload CPU in Step . . . . .	36-9
	Executing Switches . . . . .	36-10
	Results of Executing Switches . . . . .	36-10
	Taking Nonbootload CPUs Out of Step . . . . .	36-11
	Initializing Nonbootload CPUs . . . . .	36-11
	Shutting Down BCE . . . . .	36-11
	Setting Switches on SCUs . . . . .	36-11
	Rebooting BCE . . . . .	36-11
	Responding to the IOM Alarm . . . . .	36-11
	Forcing an ESD . . . . .	36-12
	Problems . . . . .	36-12
Section 37	Automatic Recovery Failures . . . . .	37-1
	To Recover the System . . . . .	37-1
	Dumping Multics . . . . .	37-1
	Performing ESD . . . . .	37-1
	Reloading MPC Firmware . . . . .	37-2
	Rebooting Multics . . . . .	37-2
	Recording the Failure . . . . .	37-3
	Problems . . . . .	37-3
Section 38	Storage System Failures . . . . .	38-1
	To Respond to an MPC Failure . . . . .	38-1
	Reloading Firmware into an MPC . . . . .	38-1
	Reinitializing Firmware in an MPC . . . . .	38-3
	To Respond to Disk Problems . . . . .	38-3
	To Move 451 Packs . . . . .	38-3
Section 39	FNP Failures . . . . .	39-1
	To Take an FNP out of Service . . . . .	39-1
	Warning Users . . . . .	39-1
	Dumping or Shutting Down the FNP . . . . .	39-1
	To Reload an FNP . . . . .	39-2
Section 40	Managing the Data Management Daemon . . . . .	40-1
	To Log in the Data Management Daemon Driver . . . . .	40-1
	To Shutdown the Data Management System . . . . .	40-1
	Problems . . . . .	40-2
Appendix A	Startup Checklists of Switch Settings . . . . .	A-1
	SCU Configuration Panel Switches . . . . .	A-2
	SCU Maintenance Panel Switches . . . . .	A-2
	CPU Configuration Panel Switches . . . . .	A-3
	CPU Maintenance Panel Switches . . . . .	A-4
	IOM Configuration Panel Switches . . . . .	A-5
	FNP DIA Switches (DN6670) . . . . .	A-7
Appendix B	Glossary . . . . .	B-1

## Illustrations

Figure 2-1	Cross Barred MPCs and Disks . . . . .	2-4
Figure 2-2	A Multics System Configuration . . . . .	2-7
Figure 3-1	Multics Directory Hierarchy . . . . .	3-2
Figure 3-2	Location of Segments and Directories on Disk Volumes . . . . .	3-4
Figure 3-3	BCE States and Commands/Events Which Change Them . . . . .	3-7
Figure 3-4	Relationships Between Major Multics Software Components . . . . .	3-17
Index	. . . . .	i-1

PART I  
INTRODUCTION

# SECTION 1

## INTRODUCTION

### HOW TO USE THIS MANUAL

This manual is divided into nine parts, each of which is divided into sections. Part I introduces you to the manual itself. It tells you how the manual is organized, and how you can make the best use of the information it contains.

Part II introduces you to the Multics system. It describes Multics hardware and software, and outlines your responsibilities for each.

Part III describes the bootload console and the initializer terminal(s). It tells you how to use them to communicate with Multics. It also gives you some instructions for performing common operator procedures associated with these devices. Finally, it explains how to deal with the many kinds of messages you'll receive on them.

Part IV explains how to bring the system up and shut the system down. It includes the procedures for powering on the machines, checking switch settings, booting BCE, booting Multics, logging in the I/O daemons and the backup daemons, shutting the system down, and powering off the machines.

Part V describes everyday operations involved in managing storage devices. It includes the procedures for dealing with tapes, user I/O disks, and storage system disks.

Part VI describes everyday operations involved in managing unit record devices. It explains how to deal with the I/O daemons that control printers, card punches, card readers, and remote devices.

Part VII describes everyday operations involved in managing the Multics backup systems. It explains how to deal with the backup daemons that control hierarchy and volume dumps and retrievals.

Part VIII describes miscellaneous everyday operations. These include: communicating with users; managing user logins, logouts, and communications channels; managing absentee jobs; setting different system modes; doing dynamic reconfiguration; and leaving notes in the system log.

Part IX describes special operations. It begins by explaining procedures for using the DPU and the DMP/VIP on a DPS 8 system. Then it tells you how to recover from various kinds of system failures. It also explains how to manage the Data Management daemon.

This manual also includes two appendixes. Appendix A is a set of checklists of switch settings. Appendix B is a glossary.

We suggest that you read through the whole manual once, to get a feel of the environment, and the kinds of things you'll be doing. Then go back and study each part. Once you understand the material presented in Parts I, II, and III, you will mostly use Part IV and Parts V through VIII. Hopefully, you will only need to use Part IX occasionally.

## MANUAL CONVENTIONS

Technical or other unfamiliar terms are *italicized* the first time they occur in the text. They are also included in the glossary (Appendix B). Conventions which are used to explain command lines are described in Section 4 under "Operator Interface."

## OTHER MANUALS OF INTEREST

If you would like to see complete descriptions of any of the Multics operations commands, you may refer to the *Multics Administration, Maintenance and Operations Commands* manual, Order No. GB64.

If you become interested in learning more about operating Multics than what is described here, or if you need more in-depth information about a given topic, you may refer to the *Multics System Maintenance Procedures* manual, Order No. AM81.



In addition, there are a number of manuals which document the major hardware modules and peripheral devices commonly used in a Multics system configuration. These manuals are listed below.

Order No.	Title
AL39	DPS/Level68 & DPS 8M Multics Processor Manual
AM46	L64/66/68 MTU0400/0500/0600 Operation
AN37	Series 60 CRU0600/1050 Card Reader Operation
AP88	Series 60 PRU1200/1600 Printer Operation
AT50	Series 60 CCU0400 Card Reader/Punch PCU0120 Card Punch Operation
AT71	L66/68 MSU0402/0451 Mass Storage Unit Operation
AY03	L66/68 MSU0500/0501 Mass Storage Unit Operation
AY34	L66/68 and DPS8 DATANET 6641/51/61/78 Operation Manual
AY83	L64/66/68 CRU0301/0501 Card Reader Operation
CB64	L66/68 MTU0610 Magnetic Tape Unit Operation
DA33	Series 6000 Equipment Operators Manual
DB28	Series 6000 MTS500 Magnetic Tape Subsystem
DC79	L66/68 Site Preparation Manual

You should keep copies of whichever of these manuals you need in the machine room, along with copies of the *Multics Administration, Maintenance and Operations Commands* and *Multics System Maintenance Procedures* manuals, and a copy of this manual.

In addition to the documentation provided by these manuals, your site may also have its own documentation, including *help files*. Help files are online segments which contain descriptions of commands, procedures, or concepts. Ask your system administrator about these. If appropriate, add a page to this manual with a list of the help files available at your site.

## PEOPLE TO KNOW ABOUT

The *Customer Services Division (CSD)* comprises Honeywell employees who maintain the hardware and software. This division used to be known as Field Engineering. CSD personnel often make use of online programs known collectively as *test and diagnostics (T & D)*. These programs allow them to run tests which help them maintain maximum system availability.

*System administrators* provide their sites with a particular Multics operating environment. They are responsible for such tasks as controlling and allocating resources (including those managed by RCP), registering projects and users, creating load control groups, setting prices on resources, setting limits on and billing for resource usage, scheduling system activities such as hours of operation, shift change times, and unattended service, describing site parameters and setting site options, and assuring system security. System administrators are responsible to upper management for the successful overall operation of the entire computer system. Thus, they set policies from which operators and system maintainers take their direction.

*System maintainers* are system programmers who configure and tune the operating system to make it comply with the special requirements of their sites. They are responsible for such tasks as backing up and recovering the system, salvaging and scavenging, analyzing crashes, balancing disks, setting up things like I/O daemons and the message coordinator, metering and tuning, and maintaining system data bases.

*Note:* there are no absolute, correct definitions of the terms "system administrator" and "system maintainer." So the definitions above may not correspond to the definitions you're used to. However, since every site makes its own particular distinctions between these terms, we must use our own definitions in this manual for the sake of consistency.

## HONEYWELL PROCEDURES VS THOSE AT YOUR SITE

This manual offers a general set of recommended procedures. We fully expect that they will be modified by each site to suit the particular needs of that site. They will be tailored and perhaps added to. You should be aware of this fact, and not be concerned if the procedures at your site differ somewhat from those presented here. In other words, don't worry about not doing things "by the book". If your system administrator or system maintainer specifies that tasks be done differently from the way we have described, always follow his or her instructions.

**PART II**  
**SYSTEM DESCRIPTION**

## SECTION 2

# HARDWARE OVERVIEW

A computer system consists of *hardware*, *software* and *firmware*. Hardware refers to all of the physical devices and electronic circuitry. Software refers to all of the *programs* that control the activities of the computer. A program is a set of instructions for solving a problem, coded in a language the computer understands. The software is said to *run* or *execute* on the hardware. Firmware is specialized software that is attached to hardware but subordinate to it. An overview of Multics hardware is presented in this section. An overview of Multics software is presented in Section 3. Firmware is described in the software overview.

### MAJOR MODULES

The major hardware modules described in this section are often referred to as *boxes* or *units*.

#### Memory and System Controller Unit (SCU)

*Memory* is the part of a Multics configuration that contains instructions and manipulated data. *SCUs* control memory. They also control and coordinate the activities of all other hardware modules. Each SCU has a specific amount of memory associated with it, divided into parts known as *store units*. SCUs interface between memory and CPUs and IOMs. (The CPU and the IOM are described next in this section.) SCUs also contain facilities which allow CPUs and IOMs to communicate with each other. In addition, SCUs contain calendar clocks. There are two kinds of SCUs: the Level 68 and the DPS 8. The primary difference between them is that on the DPS 8 SCU, certain switches and lights have been replaced with terminal displays. (See "DPS 8 vs Level 68" later in this section.) There can be up to eight SCUs in a Level 68 configuration, and up to four SCUs in a DPS 8 configuration. An SCU is often called a *system controller*. An SCU is defined in the configuration deck by a mem card.

#### Central Processing Unit (CPU)

*CPUs* are responsible for performing most of the computational processing done by the system. This means that they perform most of the arithmetic and logical manipulations of data. There are two kinds of CPUs: the Level 68 and the DPS 8. The primary difference between them is that on the DPS 8 CPU, certain switches and lights have been replaced with terminal displays. (See "DPS 8 vs Level 68" later in this section.) In addition, DPS 8 submodels have different performance rates from the Level 68, some faster, some slower. There can be up to 7 CPUs in a Multics configuration. A CPU is often called a *processor*. A CPU is defined in the configuration deck by a cpu card.

## Input/Output Multiplexer (IOM)

*IOMs* manage all of the *peripherals* connected to the system. Peripherals include the bootload console, terminals, storage devices, unit record devices, FNPs, and miscellaneous other devices. (Each of these is described later in this section, with the exception of the bootload console, which is described in Section 5.) Terminals are connected to FNPs, which are in turn connected to IOMs. Storage devices and unit record devices are connected to MPCs (described later in this section), which are in turn connected to IOMs. Managing the peripherals means that IOMs handle all transfer of data between them and memory. To help with the transfer of data, IOMs use *IOM channels*. An IOM channel is a connection between an IOM and an FNP, an MPC, or a console, over which the system can do I/O. The configuration deck (described in Section 3) specifies what channels exist and to what they are connected. There are two kinds of IOMs: the Level 68 and the DPS 8. The primary difference between them is that on the DPS 8 IOM, certain switches and lights have been replaced with terminal displays. (See "DPS 8 vs Level 68" later in this section.) There can be up to 4 IOMs in a Multics configuration. An IOM is defined in the configuration deck by an iom card.

### INFORMATION MULTIPLEXER UNIT (IMU)

*IMUs* are functional replacements for DPS 8 IOMs. They are controlled by internal microprocessors (instead of being hardwired). All switch functions for an IMU are done via a console connected to a *maintenance channel adapter* (MCA), a microprocessor in the IMU. IMU channels are called *integrated peripheral controllers* (IPCs). IPCs connect IMUs to other controllers or to peripherals, such as consoles and FNPs. There can be up to 4 IMUs in a Multics configuration; they can be mixed on the system with IOMs, for a total number of IOMs and IMUs not exceeding 4. An IMU is defined in the configuration deck by an iom card (see Section 3).

**NOTE:** throughout this manual, the term "IOM" refers to both the IOM and the IMU, unless otherwise stated.

## Front-End Network Processor (FNP)

*FNPs* are responsible for data communications. This means that they provide the physical and logical connections between the system and terminals, *networks*, and other computers. A network is a collection of hardware that provides the service of connecting many different pieces of data processing equipment, allowing them to communicate with each other. *FNPs* provide the physical connections by using *communications lines*. A communication line can be anything from a simple pair of wires in a circuit to the entire telephone system. One important use of communications lines is to connect *IOMs* with devices (like offsite printers) which are far away, since *IOMs* are designed to communicate only with devices which are nearby. *FNPs* are nearby, so devices which are far away are connected to *FNPs*, and communicate with *IOMs* via *FNPs*. (For more on this, see the discussion of remote devices in Section 24.) The physical connection between an *FNP* and an *IOM* is made via a cable called the *direct interface adapter (DIA)*. *FNPs* provide the logical connections by using *communications channels*. A communications channel defines connections. It is read from or written to. There can be up to eight *FNPs* in a Multics configuration. An *FNP* is often called a *multiplexer*, a *front-end*, a *communications processor*, or a *datanet*. "Datanet" comes from the model name of *FNPs* -- DATANET 6670. Although an *FNP* is a major module, it is often referred to as one of the peripherals. An *FNP* is defined in the configuration deck by a prph card.

## Microprogrammed Peripheral Controller (MPC)

*MPCs* control either storage devices (disk drives and tape drives) or unit record devices (printers, card punches and card readers). Some *MPCs* are *cross barred*. This means that they are connected to more than one *IOM*. The advantage of this is that if one of the *IOMs* breaks, the *MPCs* connected to it can still run, because the other *IOM* will pick up its load. An *MPC's* connection to an *IOM* is called a *link adapter*. An *MPC* can have one or two link adapters. If it has two, they are usually connected to different *IOMs*.

For an illustration of cross barred *MPCs*, see Figure 2-1.

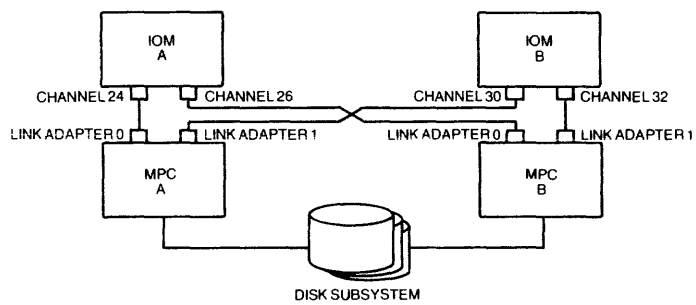


Figure 2-1. Cross Barred MPCs and Disks

## PERIPHERALS

*Peripherals* are devices which are connected to a Multics system configuration and controlled by it. Peripherals include terminals, storage devices (tape drives and disk drives), unit record devices (printers, card punches and card readers), FNP's (described earlier), the bootload console (described in Section 5), and miscellaneous other devices. They communicate with the CPU through the IOM.

### Terminals

*Terminals* are devices used to send input to the system and receive output from the system. There are basically two types of terminals: *hardcopy* or *printing terminals*, and *video* or *crt* (cathode ray tubes) *terminals*. Both have keyboards that resemble those on typewriters. A hardcopy terminal prints input and output on paper. A video terminal displays input and output on a television-like screen.

### Storage Devices

#### DISKS

*Disks* are the principal means of storing information on Multics. An individual disk unit is called a *disk pack*. The device which houses packs is called a *disk drive*. Access to disk drives is controlled by a *disk controller*. Some disk drives are *cross barred*. This means that they are connected to more than one disk controller. The advantage of this is that if one disk controller breaks, the disk drives connected to it can still run, because the other disk controller will pick up its load.

For an illustration of cross barred disks, see Figure 2-1.

Multics uses disks in two ways: as *user I/O disks* and as *storage system disks*. A user I/O disk belongs to a user. It can contain any kind of data in any format. A storage system disk belongs to the system. It contains some part of the storage system hierarchy (described in Section 3), in a standard format. One important quality of a disk pack is its capacity for shared access. Information stored on a disk pack can be used by many people at the same time (as long as they've been given proper access). However, the owner of a user I/O disk pack may be the only person authorized to use it.

You will often hear the word "volume" used in reference to disks. A *physical volume* is a set of data accessed as a group. Some disk packs contain one physical volume, while others contain two or three. A *logical volume* is a set of physical volumes which have some logical relationship to each other. For a detailed discussion of disk volumes, see "Disk Volumes" in Section 3.



There are different models of disks. The first way they differ is in whether they contain one, two, or three physical volumes and in how they are divided. The second way they differ is in whether or not the packs are demountable. The third way they differ is in how much information they can hold. The 451 disks contain one physical volume, have demountable packs and hold less information than other disks. The 500 and 501 disks contain two physical volumes divided by the MPC firmware, have packs that are not demountable, and hold more information than 451 disks. The 3380 and 3381 disks contain two and three physical volumes respectively, which are divided by the Multics file system software into *subvolumes*. These disks have packs that are not demountable and hold more information than 500 and 501 disks, 3381 disks being the largest.

#### *MASS STORAGE PROCESSOR (MSP)*

*MSPs* are devices that control disk drives. They are often called *disk controllers*. There are two kinds of *MSPs*: *MPCs* and *IPC-FIPs*. A disk *MPC* is a controller that is downline firmware loadable. A disk *IPC-FIP* is a vendor-supplied controller, designed according to the Federal Information Processing Standard (FIPS), that is connected by the *IPC-FIP* channel in the *IMU*. *MPCs* and *IPC-FIPs* are defined in the configuration deck by *mpc* and *ipc* cards, respectively.

#### *TAPES*

*Tapes* are another means of storing information on Multics. It's cheaper to store data on a tape than on a disk, but it takes longer to get data on and off a tape. An individual tape unit is called a *tape reel*. The device that houses reels is called a *tape drive* or a *tape handler*. Some tape drives are *cross barred*. This means that they are connected to more than one *MPC*. The advantage of this is that if one *MPC* breaks, the tape drives connected to it can still run, because the other *MPC* will pick up its load.

You may hear the word "volume" used in reference to tapes. A tape volume is a tape reel.

Differing amounts of information can be stored on tapes, depending on how they are recorded. A number that expresses the amount of information stored on a tape is called the tape's *density*. A tape with a density of 6250 bpi (bits-per-inch) holds much more information than a tape with a density of 1600 bpi.

#### *MAGNETIC TAPE PROCESSOR (MTP)*

*MTPs* are devices that control tape drives. They are often called *tape controllers*. There are two kinds of *MTPs*: *MPCs* and *IPC-FIPs*. A tape *MPC* is a controller that is downline firmware loadable. A tape *IPC-FIP* is a vendor-supplied controller, designed according to the Federal Information Processing Standard (FIPS), that is connected by the *IPC-FIP* channel in the *IMU*. *MPCs* and *IPC-FIPs* are defined in the configuration deck by *mpc* and *ipc* cards, respectively.

## Unit Record Devices

### *PRINTERS*

*Printers* are strictly output devices. They do not accept input. Printers provide hardcopy (paper) output in response to software programs. Sometimes they are called *line printers* because they print out information one line at a time. They are also called *high-speed printers*, because they print information at very high speeds, much faster than terminals.

## CARD PUNCHES

Like printers, *card punches* are strictly output devices. They do not accept input. Card punches provide punched card output in response to software programs.

## CARD READERS

*Card readers* transfer programs and data punched on computer cards to the central system. A necessary accompanying device to a card reader is a *keypunch machine*, a typewriter-like device with which you can type characters onto and punch holes into computer cards. There is also a device which functions as both a card reader and a card punch. This device is known as a *Combined Card Unit (CCU)*. The procedures for reading and punching cards with a CCU are the same as those used with separate readers and punches.

## UNIT RECORD PROCESSOR (URP)

*URPs* are the MPCs which control printers, card punches and card readers. They are often called *unit record controllers*.

## PANELS

All major hardware modules have at least one collection of switches and lights on them, called a *panel*. Panels are used to control the hardware and to make changes in the way the hardware operates.

For an illustration of a Multics system configuration, see Figure 2-2.

## DPS 8 VS LEVEL 68

There are two kinds of Multics systems: the original *Level 68* system and the new, improved *DPS 8* system. (DPS stands for *Distributed Processing System*.) Internally, the DPS 8 processor works differently from the Level 68 processor, but architecturally, they support the same set of instructions and registers. For users, the primary difference between the two processors is that some of the DPS 8 submodels are faster. For you, the primary difference is that DPS 8 processors, as well as SCUs and IOMs, do not have maintenance panels. The information that is provided by these panels in a Level 68 system is provided by displays on a terminal in a DPS 8 system. The exact panels which do not exist on DPS 8 boxes are as follows:

CPU: Maintenance, Test and Display Panels  
SCU: Maintenance and Display Panels  
IOM: Maintenance and Test Panels

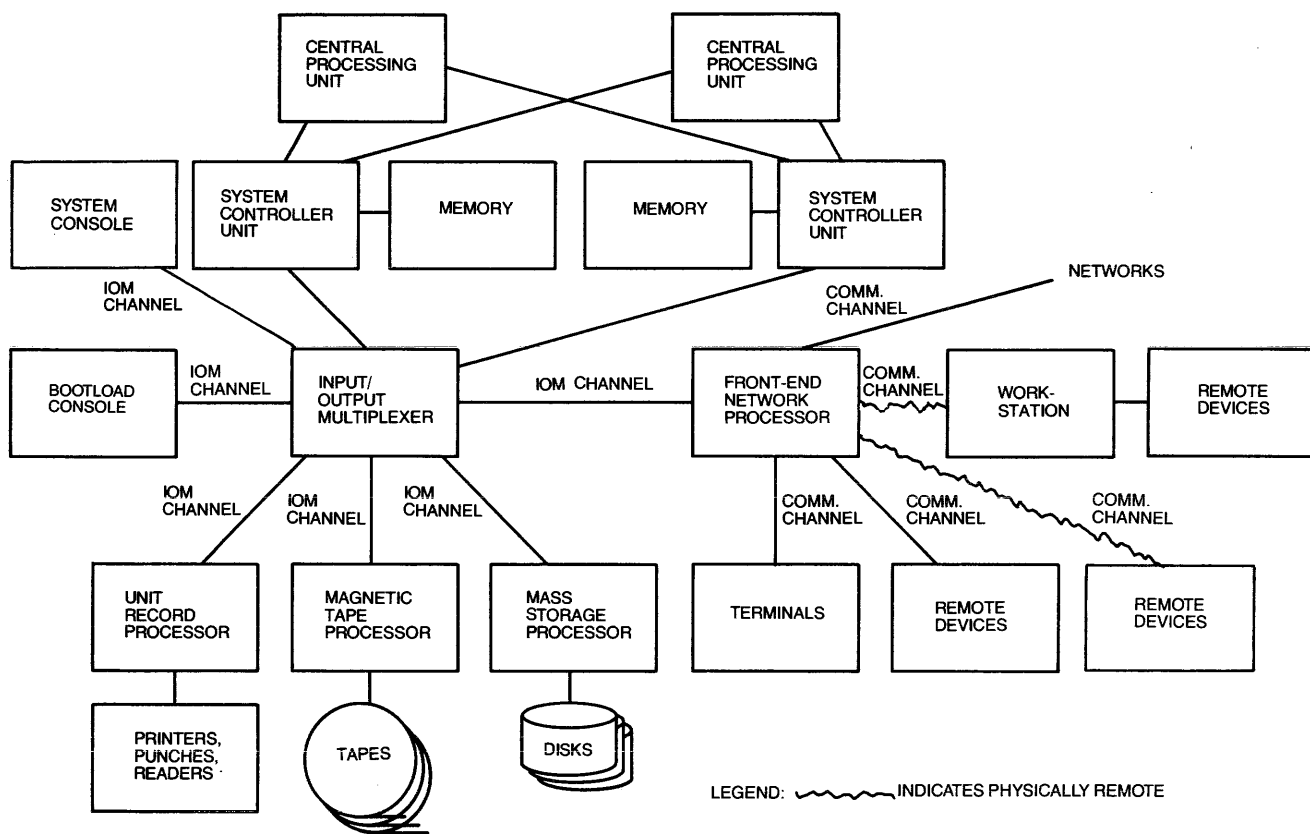


Figure 2-2. A Multics System Configuration

The displays which replace the maintenance panels are produced by the *Dynamic Maintenance Panel* (DMP), which is part of the processor. Displays from the DMP may be accessed in either of two ways: with a standard VIP terminal attached to the DMP or with a *Diagnostics Processor Unit* (DPU). The DPU serves as an interface to the DMP for the processor, the SCU and the IOM. Your decision as to whether you should use the DPU or the VIP attached to the DMP will depend on the configuration at your site. You might have one VIP and a patching mechanism to connect it to the desired DMP interface, or a separate terminal for each DMP interface, and some combination of these. You might or might not have a DPU. If you have questions about this decision, you should consult with your system administrator.

The DPU is a stand alone computer system, which has a VIP terminal attached to it (currently a VIP7205). You should not confuse this terminal, which is part of the DPU subsystem, with the terminal mentioned above, which is connected directly to the DMP.

The DPU provides a maintenance capability that includes remote maintenance control of the DPS 8 processor, SCU and IOM. In other words, the switch settings and the contents of various registers for these units may be displayed on the attached terminal.

In addition to the above-mentioned panels being replaced, the configuration panels on these three units are packaged somewhat differently than they are on the Level 68 machines. The FNP and memory have not changed with DPS 8.

For normal operations, you will find little difference between the procedures you follow for the DPS 8 and those you follow for the Level 68. However, there is a difference in some procedures for recovering from system failures. The most notable of these is executing switches to return to BCE. Differences in procedures between the DPS 8 and the Level 68 are noted throughout this manual.

## SECTION 3

# SOFTWARE OVERVIEW

### STORAGE SYSTEM

The *storage system* is the combination of hardware and software that Multics uses for storing information. It is an organized collection of *segments*, which contain *all* information, both that which belongs to the system and that which belongs to users.

Segments are the basic unit of information stored in the storage system. A segment is analogous to a file on other systems. (However, in a Multics system, the CPU can access segments directly, while other systems must use input and output instructions to read files.) A segment may contain data or programs. Certain segments which contain data used by the system are called *data bases*.

Segments are grouped into *directories*. A directory is analogous to a catalogue on other systems. It is a "catalogue" of segments and subdirectories.

Directories and segments are organized into other directories and form a hierarchy called the *storage system hierarchy*. The storage system hierarchy may be viewed as an upside-down tree with a single directory called "the root" at the top and the rest of the directories spreading out like branches from there downward. Every segment and directory has a name known as a *pathname*, which specifies its location in the storage system hierarchy (for example, ">udd>ProjectA>User1>Segmentz"). The storage system hierarchy is also called the *directory hierarchy*. For an illustration of this hierarchy, see Figure 3-1.

### DISK VOLUMES

The storage system is divided into *logical volumes* so it can be managed one piece at a time. A logical volume is made up of one or more *physical volumes*, which are sets of data accessed as groups.

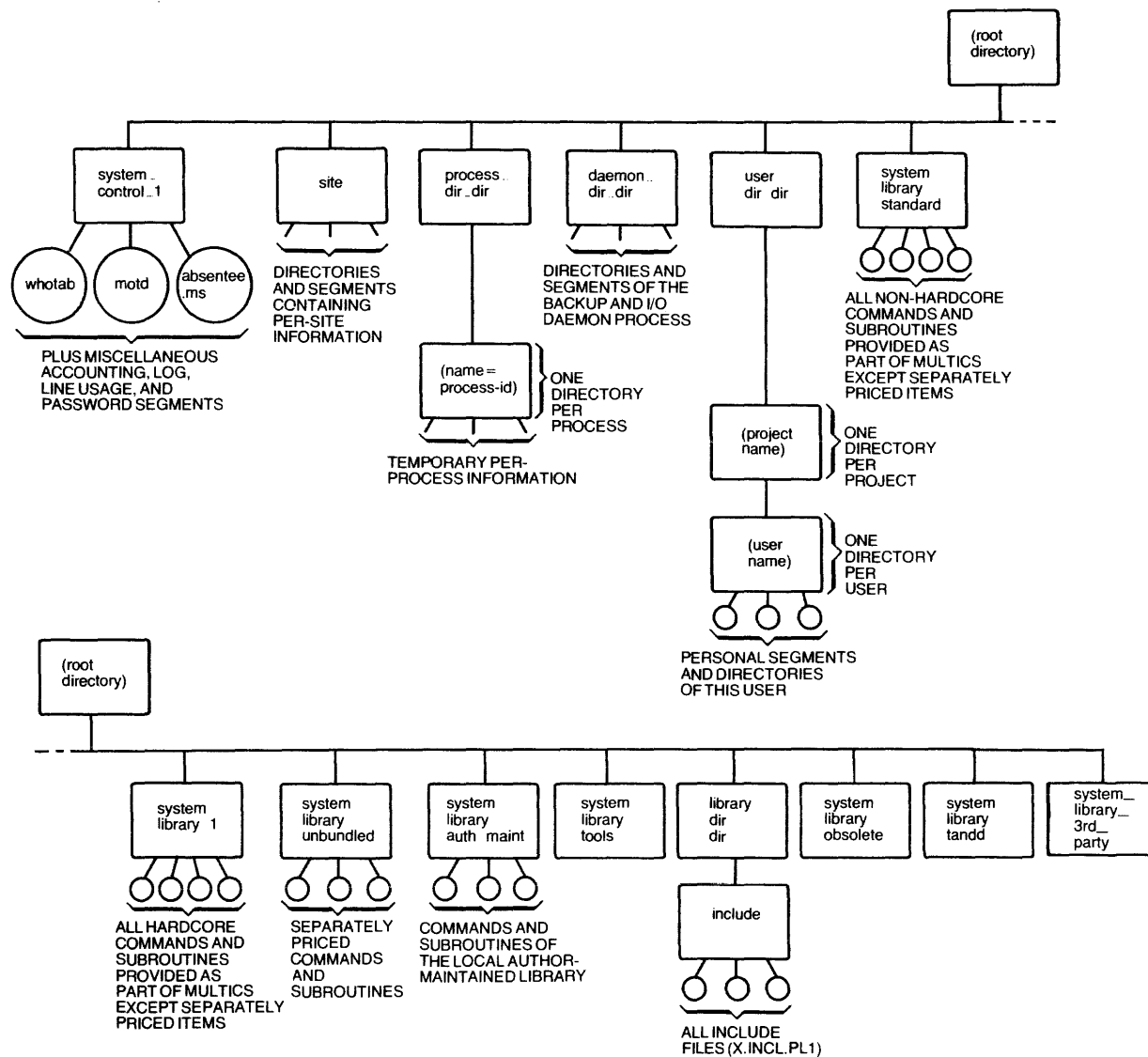


Figure 3-1. Multics Directory Hierarchy

Physical volumes are contained on disk packs. The number of physical volumes on a disk pack and how they are separated depend on what model of disk it is. Model 451 disks have only one physical volume per disk pack and therefore require no separation. Model 500/501 disks have two physical volumes per disk pack and are separated by device numbers managed by the disk controller firmware. Model 3380/3381 disks have two/three physical volumes per disk pack, respectively. They are divided into subvolumes controlled and managed by the Multics file system software. Each subvolume contains one physical volume.

The relationship between the logical volume "Public" and its constituent physical volume "pub1" is like the relationship between an encyclopedia and its Volume A. You should note that there is no fixed correspondence between the name of a logical volume and the names of the physical volumes which constitute it. However, most sites do adopt some kind of correspondence by convention. Your system administrator can tell you if there's a convention in use at your site.

When the word "volume" is used in reference to user I/O disks, it always means "physical volume." When the word volume is used in reference to storage system disks, it can mean either "physical volume" or "logical volume."

To use a logical volume, all of its physical volumes must be *mounted*. A physical volume is said to be mounted when the pack(s) are housed in the drive, the drive is operationally ready, and this has been made known to Multics.

There are two kinds of storage system logical volumes: *public* and *private*. If a logical volume is public, access to it is unrestricted, and its physical volumes are usually permanently mounted. Users don't have to request that the logical volume be mounted before they can use it. If a logical volume is private, access to it is restricted, and its physical volumes are usually not permanently mounted. Users have to explicitly request that the logical volume be mounted before they can use it. The system asks you to mount the physical volumes which constitute the logical volume. Then it allows users to use the logical volume.

When a new directory is created, its segments usually reside on the same logical volume as the segments of the directory immediately above it. An exception to this is made for a *master directory*. The segments of a master directory reside on a different logical volume than the segments of the directory immediately above it.

For an illustration of the way segments and directories are located on disk volumes, see Figure 3-2.



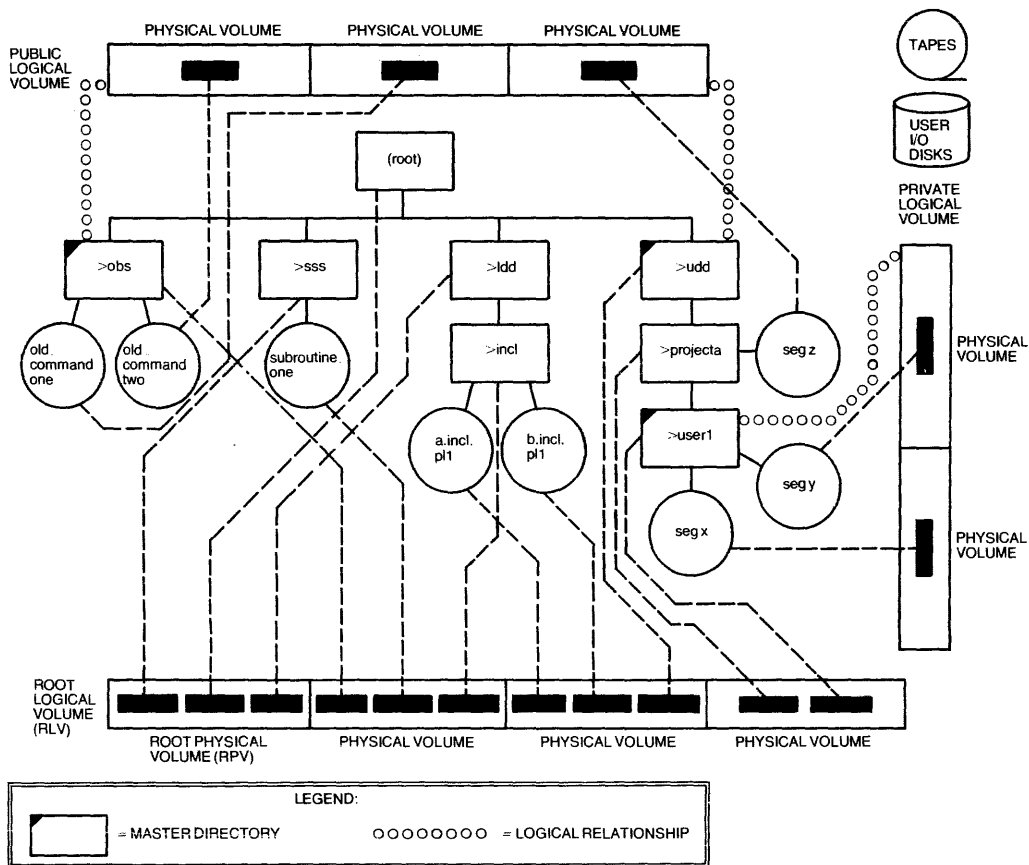


Figure 3-2. Location of Segments and Directories on Disk Volumes

There are also two special volumes known as the *root physical volume* (RPV) and the *root logical volume* (RLV). The RPV is the single physical volume required to boot BCE, bring Multics up as far as ring 1 command level, and do ring 1\* functions like reloading volumes and performing some kinds of crash recovery. The RPV is a physical volume of the RLV. The RLV is the special logical volume which contains *all* of the directories in the hierarchy, plus all of the files needed to bring Multics up the rest of the way, to ring 4 command level. (Rings 1 and 4 are described later in this section.) The information contained in the RLV is that which must be available to the system no matter what. The RLV is often called the *root*.

## OPERATING SYSTEM

An *operating system* is a set of programs which reside in a computer. The operating system interprets users' instructions, controls the hardware, and otherwise supervises the basic operation of the computer. An operating system is also responsible for sharing system resources among many users. BCE is an example of a very simple\* operating system, while Multics is an example of a very complex operating system. BCE and Multics are described later in this section. \*

## BOOTING

*Booting* (or *bootloading*) is the process of loading a set of programs (an operating system) into memory, linking them together so they can refer to each other, setting up any necessary data bases, and running the programs to start up a system. To boot Multics, you first have to boot a smaller, more primitive operating system called *BCE*. (See "Bootload Command Environment" and "Multics" next in this section.)

You may hear people use the terms *cold* and *warm* in reference to bootloads. A *bootload* is *cold* if it completely recreates the operating environment, and/or generates a new storage system hierarchy. It is *warm* if it assumes that some information from the previous *bootload* is to be used and/or maintains the current storage system hierarchy. The period of time between a Multics *bootload* and a Multics shutdown is sometimes called a *bootload session*, a *service session*, or just a *session*.

The *bootload SCU* is the SCU which is configured with a base address of zero. It is also called *low-order memory*. The *bootload SCU* must be the first SCU listed in the config deck. The *bootload CPU* and the *bootload IOM* are the CPU and the IOM selected by the *bootload SCU*. The *bootload SCU*, CPU and IOM must all be "on" in the config deck. (See "Configuration Deck" later in this section.) Instructions for setting the *bootload* boxes are given in Section 9.

\*

## BOOTLOAD COMMAND ENVIRONMENT (BCE)

*BCE* is a simple operating system. It provides a boot and crash environment for Multics. (See "Crash" later in this section.) This means that BCE must be running before Multics can be booted, and if Multics crashes, the system returns to BCE. BCE performs the following functions:

- booting Multics
  - dumping main memory and disks
  - initiating emergency shutdown (ESD) of Multics
- \* You must boot BCE in order to boot Multics.

BCE is initially contained on a tape called the *BCE/Multics system tape*. It must be booted from the tape into the system under the direction of commands that you issue at the bootload console. Once BCE is booted, it resides on the RPV disk, in a special area known as the *BCE partition*. Instructions for booting BCE are given in Section 11.

You communicate with BCE by typing *BCE commands* at the *bootload console*. Instructions for using the bootload console are given in Section 5.

BCE is always in one of four states. These states are identified by four different ready messages:

- bce (early) 0825.6:  
system has just been booted from scratch and is ready to boot BCE
- bce (boot) 0826.3:  
system is ready to boot Multics
- bce (crash) 0829.1:  
Multics has crashed
- bce (bce\_crash) 0834.7:  
BCE has crashed

See Figure 3-3 for an illustration of the BCE states and some commands/events which change them.

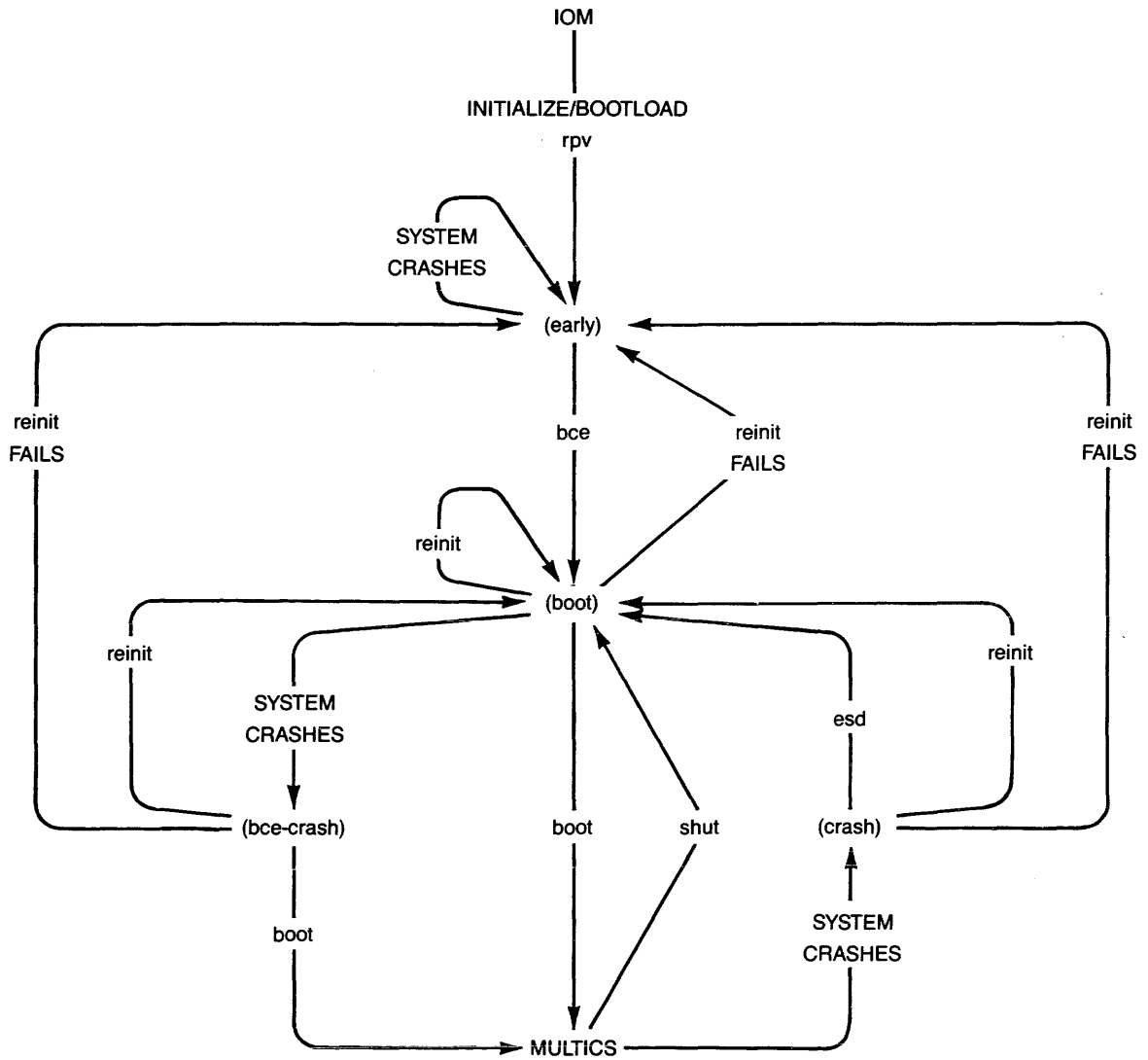


Figure 3-3. BCE States and Commands/Events Which Change Them

## MULTICS

*Multics* (Multiplexed Information and Computing Service) is a complex operating system. The programs needed to boot Multics are read from the *BCE/Multics system tape* by BCE, and stored in the BCE partition on the RPV. Thus, Multics is booted from programs stored on disk, under the direction of commands that you issue at the bootload console. Instructions for booting Multics are given in Section 13.

## SYSTEM STARTUP

*System startup* is the process of beginning Multics service. Starting up the system is also called *initialization* or *bringing the system up*.

## SYSTEM SHUTDOWN

*System shutdown* is the process of ending Multics service. There are two kinds of shut downs: *orderly shutdowns* and *crashes*. An orderly shutdown is a normal shutdown, one that you've planned. A crash is a shutdown that is the result of a system malfunction and is unplanned. (See "Crash" next in this section.)

## CRASH

A *crash* is a Multics system malfunction that causes the system to become unavailable to users. A *system crash* is an unplanned termination of system availability, caused by problems with either the hardware or the software. An *FPN crash* is an unplanned termination of service from an FNP, which causes processes connected to that FNP to be disconnected. In certain cases, deliberate (or inadvertent) actions on your part may provoke a crash.

One of the most important things you must do to respond to a crash is to take a *dump*. A dump is a "snapshot" of the state of the system. Taking a dump is like taking a picture of the system databases and user files. A dump serves as a record of the state of memory at a particular time. System maintainers use dumps to help them figure out why crashes occur. The FNP memory can be dumped as well as the main memory. Don't confuse this kind of dump with any of the backup dumps described later in this section.

## EMERGENCY SHUTDOWN

Whenever a Multics session ends, it's important to make sure that the storage system is in a consistent state. When you shut the system down normally, this happens automatically. But when the system crashes, it doesn't. *Emergency shutdown* (ESD) is a program which is run after a crash, as part of the crash recovery process. It makes the contents of the storage system consistent, as though the system had been shut down normally instead of crashing.

## CONFIGURATION

*Configuration* can be divided into three parts. The first part involves choosing hardware modules and connecting them with cables. Your site does this once when it first buys or rents a Multics system, and again each time it changes or adds new hardware modules. The second part involves setting switches on the hardware modules to connect them into an arrangement that can run Multics. You do this each time you bring the system up, before you boot BCE. The third part involves telling the software how to use the hardware: what hardware is available, how the software should set itself up to run on that hardware, and what state the hardware is in. You do this each time you bring the system up, after you boot BCE, and before you boot Multics. The information the software needs is conveyed to Multics via the *configuration deck*. The configuration deck is also called the *config deck*. The configuration deck isn't a physical deck of punched cards. It is data which you feed to the system from a tape or from the bootload console. The information in the configuration deck must correspond to the actual hardware configuration, the switch settings of the major modules, the cabling between the modules, and the operational readiness of the hardware, the peripherals, and some of the software data bases. BCE maintains its own version of the config deck. BCE passes the information in the config deck to Multics. Multics considers it to be the configuration within which it must run. When you do dynamic reconfiguration (described below), Multics updates the config deck to reflect the changes you make.

The only "cards" in the configuration deck that you'll have to deal with are *cpu, mem, iom, prph, chnl, mpc, ipc, and klok cards*. A *cpu* card identifies a processor, a *mem* card identifies a system controller, an *iom* card identifies an IOM, a *prph* card identifies a peripheral, a *chnl* card identifies an alternate path to a peripheral, an *mpc* card identifies the type of peripheral controller, an *ipc* card defines the channels as being FIPS type, and the *klok* card provides information about how to interpret the readings of the calendar clock. You must edit a *cpu*, *mem*, *iom* or *prph* card to change the state of a processor, system controller, IOM, FNP or bootload console. (For details on these operations, see Section 12.) You must edit the *klok* card to change the calendar clock for daylight savings time. (For details on this operation, see Sections 10 and 11.)

*Dynamic reconfiguration* is the process of adding or deleting a major hardware module or peripheral to or from the Multics configuration when the system is at Multics level (i.e., while the system is running). For details on dynamic reconfiguration, see Section 32.

## SYSTEM SOFTWARE

*System software* refers to the programs that come with Multics. It's divided into two parts: the *hardcore supervisor* and the *online software*.

## Hardcore Supervisor

The *hardcore supervisor* performs the supervisory functions of the system. It is usually referred to as the *supervisor*. Sometimes it's also called the *hardcore*. The supervisor is that part of the system software which can't be changed while the system is running. It includes programs which must be present to bring the system up, and programs which run the storage system. Hardcore programs are included on the BCE/Multics system tape. When the system is running, the supervisor resides on disk, in a special area known as the *hardcore partition*.

## Online Software

The *online software* is that part of the system software which can be changed while the system is running. It includes all of the programs which live in the storage system and use it.

## PROCESSES

A *process* is a user's share of the system. A user can be an actual person, such as yourself, or it can be a special part of the system, such as the initializer or a daemon. (See "Initializer Process" and "Daemons" later in this section.) A process begins when a user logs in and ends when a user logs out.

Most processes on Multics are *interactive*. This means that the user they belong to is a person using a terminal, who is exchanging information directly and immediately with the system. Some processes are *absentee*. This means that they run when the user they belong to is not logged in and not interacting with the system. Instead, the user has prepared an entire job beforehand, and is having it run at a specified time or when specified resources are available. An absentee process is analogous to a batch job on other systems. Absentee jobs are controlled by the *absentee facility*.

## INITIALIZER PROCESS

The *initializer process* is the system control process for Multics. It's the only process that can create and destroy other processes. Sometimes it's called the *answering service process*. (See "Answering Service" later in this section.) It's also called *system control* or just the *initializer*. The initializer process is created when Multics is booted and remains active as long as Multics is running. Multics can't operate without it. The initializer process performs the following functions:

- bringing the system up
- operating the answering service (i.e., "listening" for users logging in)
- servicing operator commands
- handling user requests (such as login, logout, `new_proc`)
- managing system terminals and all other terminal channels
- routing messages (such as daemon messages)
- taking care of administrative tasks for the system
- shutting the system down

You communicate with the initializer process by typing *initializer commands* at an *initializer terminal* or the bootload console. Instructions for using an initializer terminal are given in Section 6.

## RINGS 1 AND 4

Multics has eight levels of privilege, called *rings*. Most other computer systems only have two levels. Lower numbered rings are more privileged than higher numbered rings. The supervisor runs in ring 0. Most user programs run in ring 4. When you first start up the system, the initializer process enters ring 1 (the *administrative ring*). Ordinarily, there's no reason for the initializer to stop in ring 1, and it moves right away to ring 4 (the *user ring*). But sometimes you'll need it to stop in ring 1 for awhile so your system maintainer can do storage system maintenance: reloading all or part of the storage system; performing some kinds of crash recovery. In ring 1, your system maintainer is allowed to manipulate the storage system in ways which are not permitted later on, because the entire storage system isn't in use yet. When he or she is finished with storage system maintenance, you can have the initializer move to ring 4. (Your system maintainer will tell you how to stop in ring 1, then move on to ring 4.) Once here, the initializer process goes into its normal state. Certain initializer commands can only be issued when the initializer process is operating in ring 1. Others can only be used when it's operating in ring 4. Some can be used at either time.

## MESSAGE COORDINATOR

The *message coordinator* is a facility which distributes messages from the initializer process to the bootload console and other terminals. It allows the initializer process to run terminal channels, and lets the daemons run without attached terminals, sending their messages to the initializer process for disposition. Most sites run with the bootload console and one terminal. Daemon messages are routed to one or both of these devices.

The message coordinator does not handle messages from the supervisor, including RCP messages. (See "Resource Control Package" next in this section.)

## RESOURCE CONTROL PACKAGE (RCP)

*RCP* controls and allocates peripheral resources. This means that it manages the assignment of all storage and unit record devices and any disk or tape volumes you can mount on them. These devices include tapes, user I/O disks, printers, card readers, and card punches. It also produces all mount messages. RCP checks a user's access to use a device or volume automatically. If necessary, it asks you to validate that the user is permitted to mount the disk or tape volume.



## ANSWERING SERVICE

The *answering service* is an important subsystem which runs in the initializer process. (That's why the initializer process is sometimes called the answering service process.) The answering service performs the following functions:

- handling dialups, logins, logouts
- supervising system accounting
- controlling interactive and absentee users
- loading and dumping FNP's

The answering service provides your system administrator with great flexibility. For example, it allows her or him to limit the number of users (in a given group of users) who can be on the system simultaneously, to specify what time of day (i.e., during what shift) a user may log in, and to control how system resources are to be shared between users. The answering service is also known as *user control*.

## EXEC COMS

An *exec\_com* is a segment containing either a series of Multics commands to be executed or a series of BCE commands to be executed. The *auto*, *dump*, *go*, and *rtb* *exec\_coms* come with the system. They each contain a series of BCE commands to be executed. In addition, your site may write its own *exec\_coms* for use in BCE or Multics.

\*

## SYSTEM STARTUP.EC

The *system\_start\_up.ec* is an *exec\_com* which is invoked automatically when the answering service is started up. If you think of the initializer as a user, then the *system\_start\_up.ec* is analogous to the initializer's *start\_up.ec*. In addition to initializing the answering service, the *system\_start\_up.ec* may contain commands which turn on the message coordinator, log in the daemons (see "Daemons" later in this section), start up the absentee facility, load the FNP's, set the message of the day, schedule system shutdown, accept additional channels for the initializer process if it's going to operate more than one terminal, and set up the disposition of various system messages to these terminals.

## ADMIN.EC

The *admin.ec* is an *exec\_com* which is invoked any time you issue an *exec* ("x") *command*. For more information on *exec* commands, see "Operator Interface" in Section 4.

## DAEMONS

*Daemons* are system service processes. They perform such tasks as I/O device control, storage system backup, and network control. I/O daemons and Backup daemons are described in this manual. There may be other daemons to deal with at your site. If so, your system administrator will tell you about them.

Each daemon has a *User\_id* and a *message coordinator name* which is assigned by the site. The *User\_id* is a name which identifies the daemon to the system. (For more about *User\_ids*, see "The Operator Interface" in Section 4.) In this manual, the message coordinator name is referred to as the daemon's *label*. You may also hear it called a *message coordinator id* or a *source id*. A daemon's label shouldn't be confused with its *User\_id*. For example, the backup daemon which controls hierarchy incremental and consolidated dumps (described in Section 26) has the *User\_id* "Backup.SysDaemon." Your site will have a label for this daemon, which might be something like "bk" or "bkup." To reply to a question from a daemon, or issue a *daemon command*, you must use the daemon's label. To avoid confusion, daemons should be logged in with the same label every time. See "Commands and Arguments" in Section 4.

*I/O daemons* control peripheral activities like the reading and punching of cards, and the printing of output on high speed printers. They also control remote and local job entry. Commands which control I/O daemons are called *I/O daemon commands*.

There is one I/O daemon *driver process* for each local or remote device on the system, and one I/O daemon *coordinator process* which controls the driver processes.

*Backup daemons* control the Multics backup system. Commands which control backup daemons are called *backup daemon commands*. See "Backup" next in this section.

## BACKUP

*Backup* refers to the Multics backup systems. These systems ensure that user segments and directories can be recovered if they are destroyed due to system failure or user error.

### Backup Functions

The backup systems perform two functions: dumping and recovery. *Dumping* is the procedure by which the backup system searches out, selects and copies (dumps) segments and directories from the Multics storage system hierarchy onto tape. Usually, only information which has changed and may be valuable in the future is dumped. The segments and directories selected for dumping are determined by the mode -- incremental, consolidated, or complete -- in which dumping is performed. (Details on the three dumping modes are given in Section 26.)

*Recovery* is the procedure by which the backup system recovers segments and directories that have been dumped onto tape and places them back into the storage system hierarchy. Recovery consists of two operations: reloading and retrieving. *Reloading* is the global recovery of a major portion of the hierarchy when it has been damaged. If necessary, the entire contents of the storage system can be reconstructed after a major system crash so that operation of the system can resume. Reloading is the major part of recovery. It is usually performed by a system maintainer. *Retrieving* is the recovery of individual segments and directories at the request of users. It occurs during normal Multics operation. Retrieving is the minor part of recovery.

## Backup Systems

There are two major Multics backup systems, hierarchy and volume. The two systems differ in the way they locate the data they must dump. The *hierarchy system* dumps segments and directories by following the Multics directory hierarchy. This means that it skips from disk pack to disk pack. On the tapes that it produces, directories and their subordinate segments are "close" to each other.

The *volume system* dumps segments and directories by following the layout of files on disk packs. This means that it dumps segments and directories in an order that is unrelated to the storage system hierarchy. On the tapes that it produces, segments and directories that reside on the same physical volume are "close" to each other.

Another difference between the systems is that the hierarchy system produces maps, which indicate those segments and directories included in each dump cycle. These maps are needed to do hierarchy recovery. The volume system does not produce maps. Maps are not needed to do volume recovery.

Yet another difference between the systems is that the volume system manages a pool of tapes. This means that you don't have to determine which tape to use. The volume system decides for you. The hierarchy system does not manage a pool of tapes, which means that you have to tell it which tapes to use. It's important to record the names of the tapes used for hierarchy dumping. You can record them in the system log book or in a separate hierarchy dump tape log. The tape names are needed to do hierarchy recovery.

## SALVAGERS

The *salvagers* are a set of programs which do cleanup work on the storage system. They detect damage and, if possible, correct it. This damage can be caused by a crash, ESD failure, or disk device errors. Sometimes a salvager is invoked automatically as part of the system bootload. Other times it is invoked automatically when damage is discovered. It can also be invoked by you on instructions from the programming support staff.

One salvager, the *volume salvager*, can only be run when the volumes being salvaged are offline; i.e., it can't be run during normal service. This salvager is becoming obsolete, and although it may be available at your site, it isn't normally used. The volume salvager has been replaced with the *volume scavenger*. The volume scavenger, along with the *directory salvager*, can be run when the volumes being salvaged are online; i.e., they can run during normal service.

## FIRMWARE

*Firmware* is specialized software that is attached to hardware but subordinate to it. In a single machine, firmware refers to a set of programs which control the electronic circuits in that machine. Sometimes it's called the *microprogram*. Disk, tape, and unit record controllers may all have firmware. With certain models of controllers, the firmware has to be loaded into the controller before it can be used. Other modules retain the firmware even if the power is turned off.

## CALENDAR CLOCK

*Calendar clocks* are contained in SCUs. The calendar clock contained in the bootload SCU is the one used by the system. It keeps the current time, precise to the microsecond. You are responsible for setting the clock when you boot BCE, and you must set it accurately. If the setting isn't correct, serious damage to the storage system can result.

## LOAD CONTROL GROUPS

*Load control groups* are groups of users created by your system administrator. He or she can limit the number of users in a load control group who can use the system simultaneously. He or she can also limit the percentage of available resources (especially CPU computational power) which the members of a load control group can use.

## LOGGING

*Logging* is a means of keeping track of important system events.

The *syserr log* is a log of messages, called *syserr messages* or *hardcore messages*, produced by the Multics supervisor and some of the online programs. The *syserr* messages are written on a reserved area of disk called the *LOG partition*, and sometimes are also printed on the bootload console. Periodically, these messages are copied into a permanent segment in the storage system.

The *answering service log* is produced by the answering service, and contains information about all user and daemon logins and logouts.

The *IO Daemon logs* are produced by the I/O daemon driver processes, and contain information about all I/O requests. These logs may or may not be present at your site depending on how the system administrator has set things up.

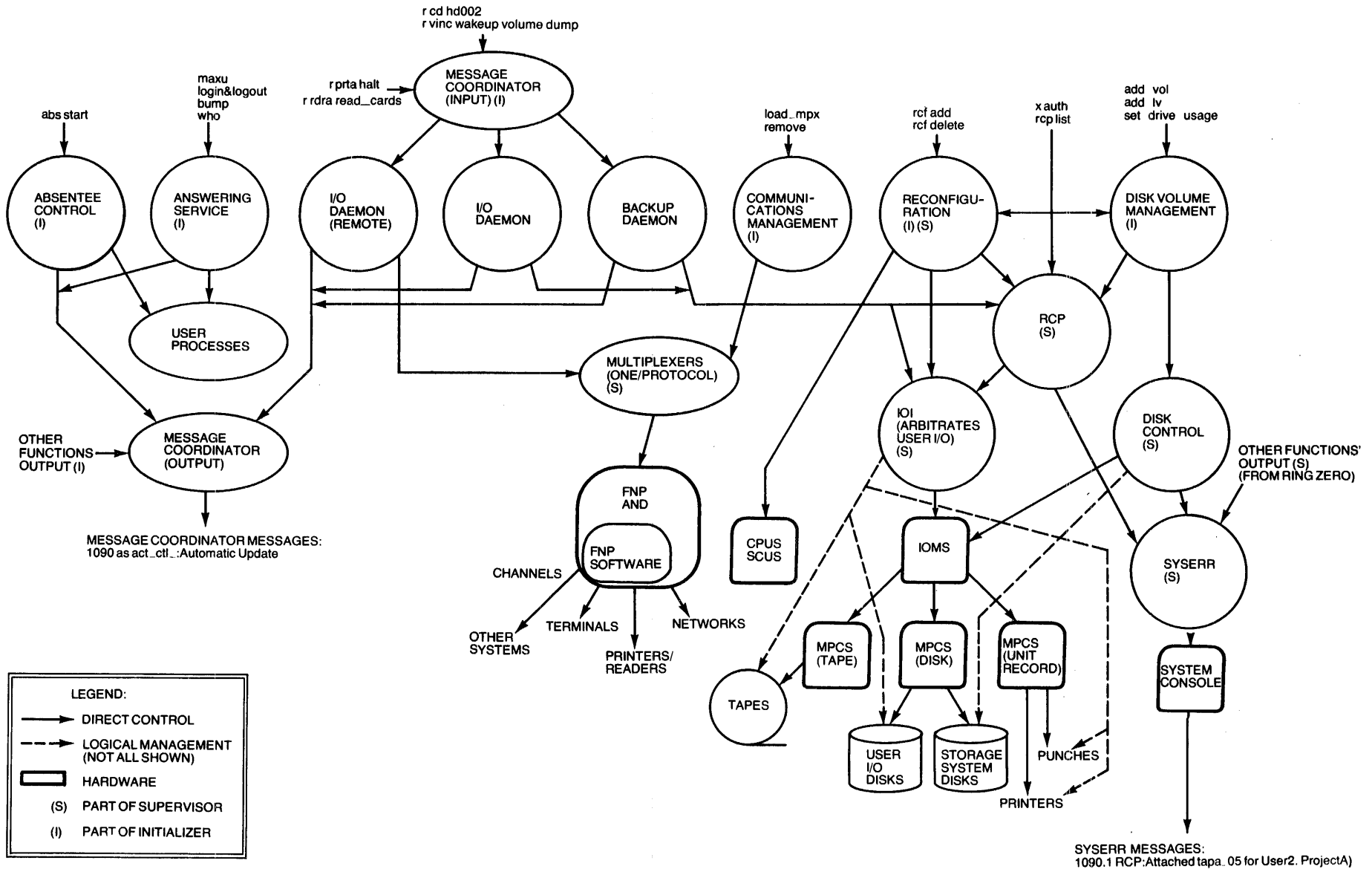
The *system log book* is a physical "pencil and paper" log. It is the operators' record of unusual occurrences. It is also a place where you can note things like the names of hierarchy dump tapes. See Section 33, "Leaving a Note in the System Log Book."

## COMPARISON OF MULTICS AND GCOS TERMS

GCOS is another large Honeywell operating system. If you have had experience working with GCOS, you may find the following list helpful. It compares some Multics software terms with their equivalent GCOS terms.

MULTICS TERM	GCOS EQUIVALENT
storage system	file system
segment	file
directory	catalogue
config deck (on Multics the config deck is a segment containing card images)	config deck (on GCOS the config deck is actually a physical deck of punched cards)
interactive process	time sharing user
absentee process	batch job
initializer	core allocator (CALC)
daemon	system program

For an illustration of the relationships between the major Multics software components, see Figure 3-4.



3-17

GB61-01

Figure 3-4. Relationships Between Major Multics Software Components



## SECTION 4

# THE OPERATOR INTERFACE

### YOUR RESPONSIBILITIES

Now that you have been introduced to the major hardware modules and peripherals, and to the most important software concepts, you are ready to learn about your responsibilities as an operator.

Your job is to control the functioning of the computer hardware by following procedures provided by your system administrators and maintainers. This job may be divided into several more specific responsibilities.

Your first responsibility is to see that users experience a minimum of service interruptions. The way system service appears to users is very important. You must see that it flows as smoothly as possible. Most of the time, this means responding promptly and correctly to requests. Requests come from the system itself, from regular Multics users, and from system administrators and maintainers.

The system will request that you perform such tasks as typing in what kind of MPCs you have and attending to troubled disk drives. Regular Multics users will request that you perform such tasks as retrieving lost segments and mounting special tapes. System administrators and maintainers will request that you perform such tasks as starting the system up and shutting it down, reconfiguring the hardware and doing complete dumps.

Your second responsibility is to see that the integrity of the storage system is preserved. Ordinarily, this means making sure that the backup dumpers used at your site run correctly and as scheduled. As part of this, you must control the backup tapes produced by the dumpers in a way that guarantees the security of the information they contain. It also means making sure that the storage system is properly shut down after a system crash, by performing an emergency shutdown.

Your third responsibility is to deal with system failures. This means responding to signs of trouble quickly and correctly. In order to respond this way, you must understand the overall configuration of the hardware at your site, the functions of BCE, and many aspects of Multics. \*



## YOUR USER\_ID

To communicate with the system, you usually use the bootload console and the initializer terminal(s) (described in Sections 5 and 6 respectively). If your site requires operator authentication, you will need a *User\_id* to sign on. (See "Operator Authentication" later in this section.) In addition, you may also use a regular user terminal at times, to issue *standard Multics commands*. If you do, you will need a *User\_id* to log in. Your *User\_id* is a name that is assigned to you when you are registered as a Multics user. It identifies you to the system. Your *User\_id* has the form:

```
Person_id.Project_id
```

in other words:

```
Yourname.Operator
```

It is assigned to you by your system administrator. When you sign on or log in as a regular Multics user, you only need to type your *Person\_id*, not your whole *User\_id*. (Your system administrator will also assign you a password, to use both when you sign on and when you log in.)

## COMMANDS AND ARGUMENTS

The interface between you and Multics consists of a set of *commands*. A command is an instruction you type on the bootload console, an initializer terminal, or a regular user terminal. It tells the system what to do or provides it with information. Many commands require one or more *arguments*. An argument is an additional piece of information that you type after a command, on the same line. Many arguments vary from site to site. Those which do are indicated as such throughout this manual. (See the discussion of command lines below.)

All of the commands described in this manual are printed exactly as you will type them. In all cases, the short form of command names is used. The long form is mentioned in the notes explaining the command line.

In this manual, most of the command lines you're supposed to type are presented in two forms. In the first form, any part of the line which you must fill in with your site's own value is represented by a placeholder. In other words, any argument or other piece of information which varies from site to site is enclosed in angle brackets, as in "<your Person\_id>." In the second form, an example is given. Those parts of the command line which were represented by a placeholder in the first form are replaced with reasonable values, as in "MacKenzie." When you type the line yourself, you should substitute your site's own values. So in this case, you should type your own *Person\_id*, *not* the sample *Person\_id* "MacKenzie" we gave you in the second form, and *not* the literal words "your Person\_id" we gave you in the first form.

If you are issuing a BCE, initializer or standard Multics command, you simply \* type the name of the command and any arguments that may be required. If you are issuing an exec command, one of the commands contained in the admin.ec, you must precede the name of the command with an "x " (short for exec). If you are issuing a command in order to reply to a daemon, you must precede the name of the command with an "r " (short for reply) and the label of the daemon. Don't worry about memorizing which commands are which. All of the instructions in this manual for accomplishing operator tasks specify when you need to type an x or an r.

## OPERATOR AUTHENTICATION

Your site may require operator authentication. If it does, then once you've brought Multics up to ring 4, you must sign on to both the bootload console and any initializer terminals you plan to use before you can enter commands on them. (You don't have to sign on when BCE is running, or when Multics is in ring 1.) Signing \* on means being responsible for all of the commands entered at that console or initializer terminal. You remain responsible until you sign off or until someone else signs on. *Note:* if your site doesn't require operator authentication, you can still sign on, if you like.

### Signing On

1. Go to the console or initializer terminal you want to sign on to. Type:

```
sign_on <your Person_id>
```

for example:

```
sign_on MacKenzie
```

2. The system will respond with:

```
Password:
```

3. Type your password.

4. If you're signing on to the bootload console, the system will respond with:

```
sign_on: MacKenzie signed on as operator on channel otw_.
```

If you're signing on to an initializer terminal, the system will respond with:

```
sign_on: MacKenzie signed on as operator on channel a.h013.
```

If someone else was signed on, it will respond with:

```
sign_on: Bongo signed off.
```

then with one of the messages above.

5. Repeat steps 1-4 on every console and initializer terminal you plan to use.

If you forget to type your `Person_id` when you sign on, the system will prompt you for it. After you type it, the system will continue as usual by prompting you for your password.

If you start to sign on, then decide you don't want to, type anything except your password. The system will not sign you on.

If you forget to sign on before you enter your first command, the system will run the `sign_on` command for you. It will prompt you for your `Person_id` and your password. After you type them, it will sign you on, then run your original command.

### Signing Off

1. Go to the console or initializer terminal you want to sign off of. Type:

```
sign_off <your Person_id>
```

for example:

```
sign_off MacKenzie
```

2. The system will respond with:

```
sign_off: MacKenzie signed off.
```

Your site may have a timeout period. If it does, you are automatically signed off if you don't enter any commands for a period of time longer than the timeout period.

Also, you are automatically signed off if someone else signs on, since only one person can be signed on to each console or terminal at a time.

### GETTING HELP

If you're in ring 4 and you need help with a command, type:

```
help <command name>
```

for example:

```
help down
```

The system will print information about the command, including its long and short names, its function, how to use it, and what arguments it requires. The system prints the information in blocks. The first block begins by telling you how many lines of information are in the first block, and how many lines of information are available in total. The first block, and all of the following blocks, end by telling you what kind of information is contained in the next block and how many lines of information are in the next block, and by asking you if you'd like to see the next block. For example, here is the first block of information the system prints about the down command:

```
>doc>subsystem>operator>down.info    (3 lines follow; 60 in info)
01/07/85 down
```

Syntax as a command: down time {back} {reason}

Function (6 lines). More help?

If you'd like to see just the next block, type "yes." If you'd like to see all of the remaining blocks, type "rest." If you'd like to skip just the next block, type "skip." And if you'd like to skip all of the remaining blocks, type "quit." The following example shows what the system prints if you type "yes" (or "y") each time it asks you if you want more information:

Function (6 lines). More help? y

Function: schedules an automatic shutdown for a specified time. It also schedules an automatic bump of users, for N minutes before the time of the shutdown. At the time of the bump, all interactive users are warned that they will be logged out in N minutes. To determine the value of N, see "Notes" below. This command can be used in ring 4 only.

Arguments (10 lines). More help? y

Arguments:

time

is the time to schedule an automatic shutdown. It is a clock reading, acceptable to convert\_date\_to\_binary. If it contains spaces, it must be enclosed in quotes.

back

is the time announced to users when the system will come back up. It must be in the same form as the time argument, above.

reason

is a message that tells users the reason for the shutdown.

Notes (20 lines). More help? y

Notes: N is figured using the value of warning\_time, a parameter in the installation\_parms segment. The warning\_time parameter in the installation\_parms segment is the number of real-time seconds between warning (of an automatic logout) and actual logout. (Refer to the Multics System Administration Procedures manual, Order No. AK50.) N equals warning\_time divided by 60, rounded up to the next whole minute. For example, if the installation\_parms segment shows a warning\_time value of 6000 seconds, then N will be 100 minutes.

When the system is started up, a check is made to see if a down command has been issued that has not yet taken effect. If one is found and more than 30 minutes remain before it takes effect, the down command is reissued automatically.

If absentee is up when the down command is issued an automatic abs stop is set up for 20 minutes before the time specified by the time argument. If absentee is not up when the down command is issued, but it is brought up later, the automatic abs stop is still set up for 20 minutes before shutdown. If absentee is already being shut down when the down command is issued, the shutdown of absentee continues.

Examples (13 lines). More help? y

Examples:

To schedule a shutdown at 5:45 a.m., and to tell the users that the system is coming back at 9:00 a.m., type:

```
down 05:45 09:00 Regularly scheduled shutdown.
```

To cancel a scheduled shutdown, type:

```
down 0
```

To display the time of the next shutdown, type the down command with no arguments.

## USING SPECIAL CHARACTERS

Some of the characters you can type in ring 4 at the bootload console or an initializer terminal have special meanings. The following characters will usually cause problems for you, so you should avoid typing them:

```
□ |
```

The following characters may make typing commands easier for you, if you use them carefully:

```
; () ""
```

You may use a semicolon (;) to enter more than one command at a time. For example, instead of typing:

```
hmu
```

then typing:

```
who
```

you may type:

```
hmu; who
```

You may use parentheses ( ) to make two command lines in which only one part of the lines is different into one command line. For example, instead of typing:

```
rcf add dv dska_01
```

then typing:

```
rcf add dv dska_02
```

you may type:

```
rcf add dv (dska_01 dska_02)
```

or even:

```
rcf add dv dska_0(1 2)
```

For another example, instead of typing:

```
remove a.h013
```

then typing:

```
attach a.h013
```

you may type:

```
(remove attach) a.h013
```

You may use double quotes (") to tell the system that you want a special character to mean something different from what it usually means. For example, if you type the following command line *without* double quotes:

```
warn * * Going down in 5 minutes (for weekend)
```

the system will send users this message:

```
Going down in 5 minutes for
```

then this message:

```
Going down in 5 minutes weekend
```

because of what parentheses usually mean. This clearly isn't what you want! But if you type the following command line *with* double quotes:

```
warn * * "Going down in 5 minutes (for weekend)"
```

the system will send users this message:

```
Going down in 5 minutes (for weekend)
```

because the double quotes tell the system that you want the parentheses to mean something different from what they usually mean.

To avoid trouble, it's a good idea to always use double quotes with any command (like warn or bump) which sends users a message.

For more information on using special characters, refer to the *Multics Programmer's Reference Manual*, Order No. AG91.

**PART III**  
**COMMUNICATING WITH THE SYSTEM**



## SECTION 5

# USING THE BOOTLOAD CONSOLE

The bootload console is special for two reasons: first, it's the only terminal you can use to issue BCE commands (in addition to all other operator commands);\* second, it's the only terminal you can use before the message coordinator is running and an FNP is loaded. The bootload console is also called the *operator console*, the *system console*, or just the *console*. When a site has more than one console, only the console which is active is called the bootload console. (For a discussion of active and inactive consoles, see Section 12.) The bootload console is *hardwired* (directly connected) to the IOM. This means that information going to and coming from the bootload console doesn't go through an FNP first. This is why it can be used before an FNP is loaded (i.e., before the answering service is up).

There are two different kinds of bootload consoles: the *CSU6001/6004* and the *CSU6601*. The 6001/6004 is a hardcopy terminal. It prints its input and output on paper. The 6601 is a video terminal. It displays its input and output on a screen. In addition, the 6601 has a printer associated with it. This printer prints out a listing which is an exact duplicate of what gets displayed on the screen. This listing should be filed in a log, and kept for a week or so. Some operator procedures differ depending on which kind of console you're using. These differences are noted throughout this manual.

Remember that if your site requires operator authentication, you must sign on before you can enter commands at the bootload console (in ring 4.) See "Operator Authentication" in Section 4.

### TYPING INPUT

The bootload console differs from a regular terminal in that it cannot accept input from you at the same time that it is printing output from the system. You have to wait until the console is ready to accept input before you type. When the console is ready to accept input, the system puts it in *input mode*.

The 6001/6004 indicates that it is in input mode by turning on its INPUT light.

The 6601 indicates that it is in input mode by printing a prompt. When the\* system is at BCE or Multics level, the 6601 prompt looks like this:

M->

Throughout this manual, examples are presented as they look on a 6601. This is because the prompt makes the examples easier to understand. (This is also because it's difficult to indicate in examples that the 6001/6004 INPUT light is on.)

Once the console indicates that it is ready to accept input, you have 30 seconds to begin typing. If you don't type something within 30 seconds, the console goes out of input mode. The console also goes out of input mode any time you start typing, then stop for more than 30 seconds.

To get back into input mode on the 6001/6004, press the REQUEST key.

To get back into input mode on the 6601, press the RETURN key.

\* If you try to type when the console isn't ready to accept input, nothing happens. The keyboard is locked and doesn't allow you to type anything.

When the console is accepting input, it's sometimes said to be *unlocked* or *in unlock mode*. (Unlock mode is the same as input mode.) When the console isn't accepting input, it's said to be *locked* or *in lock mode*.

## ISSUING COMMANDS

\* Before you can issue a command on the bootload console, you must wait for a ready message. When the system is at BCE level, a ready message usually looks like this:

```
bce (boot) 0825.6:
```

When the system is at Multics level, the way a ready message looks depends on whether or not operator authentication is required at your site, and whether or not anyone has signed on. If your site requires operator authentication (or if it doesn't, but you or someone else has signed on anyway), a ready message looks like this:

```
Ready (Operator Person_id)
```

for example:

```
Ready (MacKenzie)
```

If your site doesn't require operator authentication (and no one has signed on), a ready message looks like this:

```
Ready
```

Once you have a ready message, you can type whatever commands you need to issue (assuming the console is ready to accept input).

## Correcting Errors

If you make a typing mistake, you can correct it in one of several ways.

To erase one character, type:

#

This will erase the character which immediately precedes it. (A space counts as one character. Multiple spaces also count as one character.)

To erase two characters, type:

##

This will erase the two characters which immediately precede it. Typing "###" will erase three characters, and so on. You can continue typing #s until you erase a whole line back to the beginning. The # is called the *erase character*.

To erase a whole line, type:

@

This will erase (or "kill") everything you have typed so far, from the cursor back to the left-hand side of the screen. The @ is called the *kill character*.

On the 6001/6004, you can also erase a whole line by pressing the OPERATOR ERROR key.

On the 6601, you can also erase a whole line by typing CTL-X. (This is done by pressing the CTL key, and holding it down while you type an X.)

So, if you type the following:

```
bp@boott_3### 3 str#ar
```

the command sent to the system will be:

```
boot 3 star
```

## A Note about Numbers

Many of the commands you issue at the bootload console include numbers in their arguments. BCE and Multics assume that all numbers are decimal. The only exception to this is that BCE assumes that all numbers on config cards are octal unless they are followed by a decimal point, in which case it interprets them as decimal. You don't need to be concerned about when numbers should be octal and when they should be decimal. All of the examples in this manual show you when a number should be followed by a decimal point.

## Entering Commands

Once you've finished typing a command correctly, you must enter it, to let the system know that you are ready to have it act on your command.

To enter a command on the 6001/6004, press the END OF MESSAGE (EOM) key.

To enter a command on the 6601, press the RETURN key.

## Interrupting the System

If you enter the wrong command, or the system is printing a lot of output that you don't need to see, you may want to interrupt the system.

To interrupt the system on the 6001/6004, press the REQUEST key.

To interrupt the system on the 6601, press the RETURN key.

Interrupting the system has different results, depending on whether you're in  
\* BCE or at Multics ring 1 or ring 4 command level.

\*

### *INTERRUPTING THE SYSTEM FROM BCE*

If you're in BCE and you interrupt the system, BCE responds with this question:

abort?

You may answer with any of the responses below:

n

BCE continues with whatever it was doing (use this if you hit the REQUEST/RETURN key by accident).

y

if you were executing a subrequest of a command, BCE stops just that request from finishing; if you were executing a command, BCE stops the whole command from finishing and returns you to an executing exec\_com or BCE command level.

c

if you were executing either a subrequest of a command or a command, BCE stops the whole command from finishing and returns you to an executing exec\_com or BCE command level.

a

if you were executing a subrequest of a command, a command, or an `exec_com`, BCE stops it from finishing and returns you to BCE command level; if you were receiving output, BCE stops it from being sent and returns you to BCE command level.

### INTERRUPTING THE SYSTEM FROM MULTICS

If you're at ring 1 or ring 4 command level and you were executing a command, Multics lets it finish. If you were receiving output, Multics stops it from being sent and prints this message:

```
output discarded
```

### Commands You Can Issue at the Bootload Console

In addition to BCE commands, you can also issue `initializer`, `exec`, `daemon`, and \* standard Multics commands at the bootload console. However, the procedure we recommend is to type these commands at the initializer terminal(s). Just remember that the bootload console is the *only* place where you can issue BCE commands. \*

### MESSAGES ON THE BOOTLOAD CONSOLE

Once startup is complete, the only messages which get printed on the bootload console are RCP messages, disk error messages, and sometimes salvager messages. Other messages get printed on the initializer terminal(s). (In small configurations which don't operate with an initializer terminal, all messages get printed on the bootload console.) See Section 7, "Dealing with System Messages," for more details.

### THE AUDIBLE ALARM

The bootload console contains an *audible alarm*, also called a *beeper*. Certain messages, such as RCP mount messages, will make it sound.

To turn off the audible alarm on the 6001/6004, press the ATTN/RESET key.

To turn off the audible alarm on the 6601, press any key. (The resulting key-stroke is discarded. So if you press the RETURN key to turn off the beeper, you have to press it again to get into input mode.)

### REROUTING BOOTLOAD CONSOLE ACTIVITY TO A REGULAR USER TERMINAL

You should only reroute bootload console activity if your system administrator or system maintainer asks you to.

1. At a regular user terminal on which no one is logged in, type:

dial system

2. The system will respond with:

ROSY none a.h002 dialed to Initializer.

where "a.h002" is the name of the terminal channel of the regular user terminal. Take note of this name -- you'll need it for the next two commands.

3. At an initializer terminal, type:

accept a.h002

4. The system will respond with:

channel a.h002 attached by Message Coordinator.

This means that the regular user terminal has been added as an initializer or message coordinator terminal.

5. At the initializer terminal, type:

substty otw\_ a.h002

All bootload console activity (except RCP and syserr messages) will now be rerouted to the regular user terminal.

## COMMUNICATING WITH THE MCA(S)

If your site has one or more IMUs in its configuration, you will need to communicate with the maintenance channel adapter(s) (MCA(s)). To do this, you must use the master console. The master console may be the same console as the bootload console, or it may be a different console. This master console is defined by the MCA internal configuration file, and must be one of the consoles configured in the Multics configuration.

1. To send a command to an MCA, type:

#NN>command line

for example:

#01>time 111485,120000

where "#" indicates that you want to communicate with an MCA, "01" is the number of the MCA, ">" indicates that this is an input message, and "time..." is the MCA command line.

*Note:* You don't actually type the ">"; the console prints it.

2. A message from an MCA will look like this:

```
#NN<message
```

for example:

```
#01<Monday November 14, 1985. 11/14/85 (12:00:00)
```

where "#" indicates that this message is from an MCA, "01" is the number of the MCA, "<" indicates that this is an output message, and "Monday..." is the MCA message.

## PROBLEMS

1. If the bootload console stops working, the first thing you should do is check the printer to see if the paper is jammed. If it is, clear it, and the console should start up again.
2. If the paper in the printer is OK, the console hardware may be broken. Use a regular user terminal to try the procedure described in steps 3-10 below. *Note:* if operators are not allowed to use the sac command at your site, you will not be able to perform this procedure.
3. At the regular user terminal, log in as a regular user by typing:

```
l <your Person_id>
```

for example: -

```
l MacKenzie
```

where "l" is the short form of the login command.

4. The system will respond with:

```
Password:
```

5. Type your password. The system will respond with:

```
Mackenzie Operator logged in 06/01/83 0921.4 est Wed  
from VIP7801 terminal "none".  
Last login 03/18/83 0726.2 est Fri from VIP7801  
terminal "none".  
r 09:22 3.0 3.50
```

where "r 09:22 3.0 3.50" is a ready message.

6. Type:

```
sac set_system_console OPCx -state on
```

where "OPCx" should be replaced by the name of the operator console whose operation is being restored (e.g., opca).

Then type:

```
sac set_system_console -reset
```

\* *Note:* use of this command requires that your system administrator give your User\_id special privilege.

7. The system will respond with:

```
Console opca reset for Initializer.SysDaemon
```

where "opca" is the name of the console as it appears in the config deck.

8. Log out as a regular user by typing:

```
logout
```

9. The system will respond with:

```
MacKenzie.Operator logged out 06/01/83 0928.3 est Wed
CPU usage 13 sec, memory usage 1.8 units, cost $0.28
hangup
```

10. If this procedure doesn't work, console activity will be automatically rerouted to another console (if your site has more than one) or to an initializer terminal.

### SUMMARY OF BOOTLOAD CONSOLE KEYS

FUNCTION	6001/6004 KEYS	6601 KEYS
gets the console to accept input	REQUEST	RETURN
erases characters one at a time during multics input	#	#
erases a whole line	@ OPERATOR ERROR	@ CTL-X
enters a command	EOM	RETURN
interrupts the system	REQUEST	RETURN
turns off the alarm	ATTN/RESET	any



## SECTION 6

# USING AN INITIALIZER TERMINAL

An initializer terminal is a regular terminal you can use to issue all operator commands *except* BCE commands. It is connected to the front-end of the system (the FNP). Sometimes an initializer terminal is called a *message coordinator terminal*. Initializer terminals come in a variety of models, but operator procedures are basically the same on all of them. Your system administrator defines initializer terminals by means of the `system_start_up.ec` or the `accept` command.

Remember that if your site requires operator authentication, you must sign on before you can enter commands at an initializer terminal (in ring 4.) See "Operator Authentication" in Section 4.

### ISSUING COMMANDS

Before you can issue a command at the initializer terminal, you must wait for a ready message. If your site requires operator authentication (or if it doesn't, but you or someone else has signed on anyway), a ready message looks like this:

```
Ready (Operator Person_id)
```

for example:

```
Ready (MacKenzie)
```

If your site doesn't require operator authentication (and no one has signed on), a ready message looks like this:

```
Ready
```

### Correcting Errors

You correct typing errors on the initializer terminal the same way you correct them on the bootload console. That is, by using the erase (#) and kill (@) characters. See "Correcting Errors" in Section 5.

### Entering Commands

You can enter a command on the initializer terminal by pressing the CARRIAGE RETURN (CR) key.

## Interrupting the System

If the system is printing a lot of output that you don't need to see, you may want to interrupt the system.

To do this, press the CR key. The system will respond with:

OPER:

The system will finish executing the command, but it will stop sending you output for 60 seconds. While the output is suspended, you may issue any commands you wish. There is no way to make the system stop executing the command or discard the output.

## Commands You Can Issue at the Initializer Terminal

- \* You can use the initializer terminal to issue all commands *except* BCE commands. This includes initializer, exec, daemon, and standard Multics commands. The procedure we recommend is to type these commands at the initializer terminal(s).
- \* Just remember that the bootload console is the *only* place where you can issue BCE commands.

## MESSAGES ON THE INITIALIZER TERMINAL

The initializer terminal gets messages from the initializer, the backup system, the answering service, the I/O daemons, and users. See Section 7, "Dealing with System Messages," for more details.

## MULTIPLE INITIALIZER TERMINALS

When Multics is first booted, and the initializer process is working in ring 1, it performs its input/output on the bootload console. Once the initializer process moves to ring 4, its I/O may be moved to an initializer terminal. If it is, the name of the initializer terminal's channel is printed in one of the messages you receive when you boot Multics. The message looks like this:

```
a.h004 attached by Message Coordinator
```

The initializer process can run multiple initializer terminals. If additional channels are desired, your site can arrange to have them added to the initializer during startup, by commands which run as part of the `system_start_up.ec`.

If more than one channel is connected, the output from various sources can be routed to divide the work among several terminals. For example, you may have one terminal for answering service output, where login and logout messages will be printed; one terminal for the daemons, where messages like print requests will be printed; and one terminal for you to use as a regular user terminal, where you can do things like read your mail.

When there is more than one initializer terminal, you can issue commands from any one of them (e.g., you can reply to a daemon on any one of them). Direct responses to commands you issue appear on the terminal from which you issue them. Other system messages may be routed to other terminals.

## THE AUDIBLE ALARM

The initializer terminal contains an *audible alarm*, also called a *beeper*. Messages which require your attention will make it sound.

The audible alarm on the initializer terminal works differently from the audible alarm on the bootload console. When a message which requires your attention gets routed to the initializer terminal, the following things happen:

- the system prints a line of ten asterisks (\*)
- the audible alarm beeps a number of times
- the system prints the message which requires your attention
- the system prints another line of asterisks

The audible alarm on the initializer terminal turns itself off. You don't have to do anything to stop it.

## ADDING A REGULAR USER TERMINAL TO THE INITIALIZER DYNAMICALLY

You should only add a regular user terminal to the initializer when there's a problem with one of the initializer terminals and you need to replace it.

1. At the regular user terminal you want to add (on which no one should be logged in), type:

```
dial system
```

2. The system will respond with:

```
TN300 405 a.h003 dialed to Initializer.
```

where "a.h003" is the name of the terminal channel of the regular user terminal. Take note of this name -- you'll need it for the next command.

3. At an initializer terminal, type:

```
accept a.h003
```

4. The system will respond with:

```
channel a.h003 attached by Message Coordinator.
```

This means that the regular user terminal has been added to the system.

5. You may now use the added terminal to issue operator commands.

### DROPPING A TERMINAL DYNAMICALLY

1. At an initializer terminal, type:

```
drop <terminal channel>
```

for example:

```
drop a.h003
```

where "a.h003" is the name of the terminal channel of the terminal you want to drop.

2. The system will respond with:

```
Ready (MacKenzie)
```

3. Then, on the terminal you're dropping, the system will respond with:

```
please reissue dial command
```

### REROUTING INITIALIZER MESSAGES FROM ONE TERMINAL TO ANOTHER

If an initializer terminal breaks down and needs to be removed from the system for servicing, you can reroute its messages to another terminal. This is known as *swapping ttys*.

1. If you're going to reroute the messages to a terminal which is already receiving messages, go on to step 2. If you're going to reroute the messages to a new terminal, follow the procedure described earlier in this section under "Adding a Terminal to the Initializer Dynamically." Then proceed with step 2.

2. Type:

```
substty <bad terminal channel> <good terminal channel>
```

for example:

```
substty a.h007 a.h026
```

where "a.h007" is the channel of the terminal that is broken down, and "a.h026" is the channel of the terminal to which you want to reroute the messages.

3. The system will respond with:

Ready (MacKenzie)

4. Type:

remove a.h007

5. The system will respond with:

a.h007 removed

### PUTTING TABS BACK ON A TN300

If the initializer terminal you're using is a TermiNet 300 (model number TN300), certain errors may cause the tabs to be cleared. You'll know this has happened if the output on the paper is all run together instead of spaced out correctly. You can reset the tabs by typing:

```
x rt
```

where "rt" is the short form of the "reset\_tabs" command.

This command should *only* be used if the terminal is a TN300.



## SECTION 7

# DEALING WITH SYSTEM MESSAGES

You will receive many messages from the system. The majority of these messages are simply intended to give you information about system activities and minor malfunctions. For the most part, you don't have to respond to them. A smaller number of the messages request some action from you. Any message that requires immediate action will make the audible alarm on the bootload console or the initializer terminal go off: a request for a tape mount, a printer out of paper, a disk drive in trouble. Some messages are repeated by the system until you respond to them.

The following kinds of messages will usually appear on the bootload console:

- BCE messages
- syserr messages:
  - RCP messages
  - disk error messages
  - salvager messages

\*

The following kinds of messages will usually appear on the initializer terminal(s):

- message coordinator messages:
  - backup daemon messages
  - I/O daemon messages
  - login, logout, and other answering service messages
- initializer command responses

The above messages are described briefly in this section. The ways you should respond to them are detailed throughout the manual.

### THE FORM OF A SYSTEM MESSAGE

A system message usually begins with the time it was sent. (Times have been included in the message examples in this section, but not elsewhere in the manual.) After the time, the message usually gives its source. After the time and the source comes the body of the message.

There are four major kinds of messages. A *BCE message* is not indented and does not include a time or a source. (The source is always BCE.)

A *syserr message* is also not indented. It includes a time with a decimal point. Its source is either the name of a subsystem (for example, RCP) or the name of a program (for example, disk\_control)

A *message coordinator message* is indented one space. It includes a time without a decimal point. Its source is either a daemon (indicated by the daemon's label) or the answering service (indicated by "as").

An *initializer command response* is a response to an initializer command. It is not indented, and does not include either a time or a source. (The source is always the initializer process.)

All of the messages described in this manual are printed exactly as the system types them.

\*

## BCE MESSAGES

BCE messages are produced by BCE. An example of a BCE message is:

```
Booting t610 on IOM a chn 14 with m610 rev.11 firmware.
```

BCE messages are only produced when the system is at BCE level -- during startup and after certain system failures.

## SYSERR MESSAGES

### RCP Messages

RCP messages are produced by RCP. An example of an RCP message is:

```
1420.1 RCP: Authenticate tapa_05. It has no label.
```

The most common RCP messages provide instructions for performing tape and disk mounts.

### Disk Error Messages

Disk error messages are produced by the supervisor. An example of a disk error message is:

```
0822.5 disk_control: dska_04 requires intervention.
```

Disk error messages tell the operator when a disk drive is in trouble, and may provide information that will help him solve the problem.



## Salvager Messages

Salvager messages are produced by the salvagers. Examples of salvager messages are:

```
0643.8 scavenger: Begin scavenge of dska_01 by
                  Scavenger.SysDaemon.z

0643.8 scavenge_volume: Freed 72 VTOCEs on dska_01.

0643.9 scavenge_volume: 9 VTOCEs on dska_01 damaged.

0643.9 scavenger: Scavenge of dska_01 by Scavenger.SysDaemon.z
                  completed.
```

Whenever a salvager is invoked to correct a physical volume, the operator will receive a whole set of messages. These messages provide information. They don't require a response.

## MESSAGE COORDINATOR MESSAGES

### Backup Daemon Messages

Backup daemon messages are produced by the volume and hierarchy backup systems. An example of a backup daemon message is:

```
1955 cd2 Input tape label
->cd2
```

A message from a daemon that begins with a "->" is called a *sentinel*, and indicates that the daemon wants input.

### I/O Daemon Messages

I/O daemon messages are produced by the I/O daemon processes. An example of an I/O daemon message is:

```
1956 prt b prt b driver ready at 06/01/83 1956.1 est Wed
->prt b
```

Again, the line beginning with a "->" indicates that the daemon wants input.

### Login and Logout Messages

Login and logout messages are produced by user and system processes when they login and logout. An example of an interactive user login message is:

```
1719 as LOGIN User1.ProjectA int a.h026.001 (create)
```

An example of an absentee user login message is:

```
1719 as LOGIN User4.ProjectB Q 3 abs2 (create) [my_absentee]
```

An example of a system process login message is:

```
1719 as LOGIN Backup.SysDaemon dmn bk (create)
```

An example of an interactive user logout message is:

```
1805 as LOGOUT User1.ProjectA int a.h026.001 0:17 $6.92 (logout)
```

An example of an absentee user logout message is:

```
1805 as LOGOUT User4.ProjectB Q 3 abs2 0:09 $3.41 (logout)
```

An example of a system process logout message is:

```
1805 as LOGOUT Backup.SysDaemon dmn bk 5:28 $23.45 (logout)
```

A great number of login and logout messages are produced. When they're concerned with interactive and absentee users, they simply provide information, and don't require a response. When they're concerned with system processes, the operator should take note of them. They may mean that a daemon is logging in to perform a task that may require operator assistance, or possibly that a daemon is logging out because it's in trouble.

### Other Answering Service Messages

Most other answering service messages simply provide information. An example of another answering service message is:

```
0345 as act_ctl_: bumping User4.ProjectB for inactivity.
```

### INITIALIZER COMMAND RESPONSES

Initializer command responses are produced by the initializer process. An example of an initializer command response is:

```
reconfigure: CPU a is now running.
```

An initializer command response is always printed on the terminal that was used to issue the initializer command.

### ERROR MESSAGE DOCUMENTATION

Every release of Multics includes an online error messages segment. This segment documents all of the error messages that can be generated by the system. It also documents quite a few messages which are not error messages, but just regular messages. The messages are listed in alphabetical order. Each message description tells you where the message gets printed, when you are likely to receive it, what it means, and what action, if any, you should take to respond to it.

If you receive an error message that you don't recognize or don't know how to respond to, you can look it up in this segment. The name of the segment is:

```
>doc>MR12.0>error_messages.doc
```

where "MR12.0" is the number of the Multics release under which you are running. We strongly recommend that you dprint a copy of this segment and keep it in the machine room.



PART IV  
BRINGING THE SYSTEM UP AND  
SHUTTING THE SYSTEM DOWN

## SECTION 8

# POWERING ON THE MACHINES

Obviously, machines must be turned on before they can be used. Most sites leave their machines powered on all the time, so you probably won't have to do this very often. If your machines are already powered on, skip to the next section.

### TO POWER ON THE MACHINES

1. Press the POWER ON button on each CPU. On a DPS 8 box, make sure the CABINET SHUTDOWN button isn't pushed in.
2. Press the POWER ON button on each SCU. On a DPS 8 box, make sure the CABINET SHUTDOWN button isn't pushed in.
3. Press the POWER ON button on each IOM. On a DPS 8 box, make sure the CABINET SHUTDOWN button isn't pushed in. On an IMU, verify that the production diskette (PROD) is in diskette drive 0.
4. Press the POWER ON button on each FNP.
5. Press the POWER ON button on each tape MPC. Push the INITIALIZE button, and then the START button.
6. Press the POWER ON button on each disk MPC. Push the INITIALIZE button, and then the START button.
7. Press the SUBSYSTEM POWER OFF button on the 3880 storage director. If the power sequencer cables are disconnected, press the SUBSYSTEM POWER OFF button on the 3380/3381 device controller.
8. Make sure the UNIT EMERGENCY switch on the 3380/3381 device controllers and the 3880 storage directors are set to POWER ENABLE. Press the SUBSYSTEM POWER ON button on each storage director. If the power sequencer cables are disconnected, press the SUBSYSTEM POWER ON button on each device controller.
9. Press the POWER ON button on the MTU8200 head-of-string device. (This button is located in the upper right hand corner of the FE panel.) Insert the floppy diskette into the diskette reader. Make sure the HEART BEAT and IDLE loop indicators come on.
10. Lift up the breaker on the back side of each tape drive. Then press the POWER ON button on the front of each drive (except 500 and 610 drives).

11. Press the POWER ON button on each 500 and 501 disk drive. Wait at least 15 minutes (CSD recommends 60), then push the START button to cycle each disk up. *Note:* if you fail to observe this procedure, you may cause damage to the disks.
12. Lift up the breaker on each 451 disk drive. The breaker is located on the bottom left of the back side of the drive. Determine if the proper packs are mounted, then push the START button to cycle the disks up.
13. Lift up the breaker on the back of each printer. Then, on each PRU1200/1600 printer, press the POWER ON button on the front.
14. If the bootload console is a 6001/6004, press the POWER ON button, then the ONLINE button. If it's a 6601, slide the POWER switch on the back of the screen to ON, and push the breaker on the back of the Rosy printer over.

### GOOD IDEAS

1. Add a page to this manual detailing the procedures at your site for dealing with lights, air conditioning, motor generators, breakers, etc.
2. Add a page to this manual with a map of your computer room. Number all of the machines. Add another page listing the machines by number, giving their names, model numbers, etc.

## SECTION 9

# CHECKING SWITCH SETTINGS

Switches are set on the major hardware modules to connect them into an arrangement that can run Multics. Some sites change their switch settings all the time. Other sites set them once, and never change them again. There are rules about which switches can be set and when. These rules must be followed carefully. It's very important that the switches be set correctly before the machines are run. It's especially important that the switch settings match the information in the configuration deck.

### TO CHECK THE SWITCH SETTINGS

1. Find the checklists in Appendix A of this manual.
2. Compare the switch settings in the checklists with the switch settings on the machines. Any switches which are different on the machines should be changed to match the switches in the checklists. There may be differences between the settings in the checklists and the settings at your site. If so, you should follow your site procedures, not ours.
3. If you need to see pictures of the panels containing the switches, refer to the *Multics System Maintenance Procedures* manual, Order No. AM81.

### TO SET THE BOOTLOAD IOM, SCU AND CPU

Your system administrator will tell you which IOM, SCU and CPU are the bootload boxes. You will usually use the same IOM, SCU and CPU as the bootload boxes every time you boot the system. If you're supposed to use a different box for some reason, your system administrator will let you know.

If you've followed the procedure above for checking switch settings, the bootload boxes should already be set correctly. In that case, you can use the steps below as a double check.

1. Go to the bootload IOM. Set switches on the config panel as follows: set the PORT ENABLE switch which corresponds to the bootload SCU to ON; set the ASSIGNMENT switch which corresponds to the bootload SCU to 000. Note that there are no switches on an IMU model IOM. The SCU definition is contained in the internally-maintained IMU configuration file.
2. Go to the bootload SCU. Set switches on the config panel as follows: set the PORT ENABLE switches which correspond to the bootload CPU and the bootload IOM to ON; set the MASK A MASK/PORT ASSIGNMENT switch to the number of the SCU port to which the bootload CPU is connected. *Note:* this is *NOT* the number of the bootload CPU.)



3. Go to the bootload CPU. Set switches on the config panel as follows: set the PORT ENABLE switch which corresponds to the bootload SCU to ON; set the ASSIGNMENT switch which corresponds to the bootload SCU to 000.

## GOOD IDEAS

1. Make copies of the Appendix A checklists and tape them to the machines, noting any site differences. This will make checking the settings a lot easier.
2. Have CSD make up a list of all the SCUs, CPUs, IOMs and MPCs, and their connections to each other. The list should include:
  - the SCU port number to which each CPU is connected
  - the SCU port number to which each IOM is connected
  - the CPU port number to which each SCU is connected
  - the IOM port number to which each SCU is connected
  - how big the memory in each SCU is
  - the values of the store size patch plugs for each CPU and IOM port on each L68 SCU
  - the IOM channel to which each MPC is connected
  - the MPC link adapter to which each IOM is connected
  - the model number of each MPC
3. Label all switches used by CSD for T & D with their Multics positions. On the IOM, these switches are as follows:
  - configuration panel:*
    - IOM BASE ADDRESS
    - INTERRUPT BASE ADDRESS
  - bootload panel:*
    - SOURCE on L68; BOOT SOURCE on DPS 8
    - CHANNEL NUMBER CODE on L68; CHANNEL SELECT on DPS 8
    - OPERATING MODE
  - test panel:*
    - OPERATION SELECTOR
    - STORE CONTROL
    - STOP CONDITIONS
    - AUTOMATIC RATE

On the L68 CPU, these switches are as follows:

- maintenance panel:* • ADDRESS STOP
- display panel:* • EXECUTE PB/SCOPE REPEAT  
• INITIALIZE CLEAR/INITIALIZE CONTROL
- test panel:* • FAULT CONTROL  
• TEST CONTROL

(On the DPS 8 CPU, the maintenance, display and test panels are not present.)

4. Always check the settings after CSD has done any kind of maintenance.
5. At some sites, especially large ones, some switches are changed all the time. If this is true at your site, post a list of these switches. Include all possible settings for each switch, and instructions for choosing the right one.



# SECTION 10

## BOOTLOADING BOS

The information that was in this section is obsolete and has been deleted.



## SECTION 11

### BOOTLOADING BCE

BCE must be booted when the machines have been powered off, and any other time the system is brought up from scratch.

#### TO BOOT BCE FROM SCRATCH

##### Initializing BCE

1. Mount the BCE/Multics system tape on the correct tape drive.
2. If the tape is mounted on a 500 tape drive, ready it by pressing the LOAD button, waiting for the BOT light to go on, then pressing the READY button. If the tape is mounted on a 600, 610 or 630 tape drive, ready it by pressing the START button. If the tape is mounted on an 8200 tape drive, ready it by pressing the START button, and continue with step 8. If tape does not load, go to step 3.
3. Go to the tape MPC which is connected to the tape drive on which you have mounted the BCE/Multics tape. Set switches 5, 6, 7, and 8 on the MPC configuration panel to the number of this BCE/Multics tape drive. The switches must reflect the *binary* representation of the number. For example, if the tape is mounted on drive 6, switch 5 must be set to 0 (↓), switch 6 to 1 (↑), switch 7 to 1 (↑), and switch 8 to 0 (↓). to the chart at the end of this section for the binary representations of numbers 1 through 15.
4. You must also set an MPC switch to reflect the number of the MPC link adapter that is connected to the IOM through which you are going to load the MPC firmware. (Loading firmware is described later in this section.) When you are booting BCE, this IOM must be the bootload IOM. If you are using a 500 or 601 tape MPC, the switch to set is number 2 on the MPC configuration panel. If you are using a 610 or 611 MPC, the switch to set is number 10. If the IOM is connected to MPC link adapter 0, set the switch to 0 (↓). If it is connected to link adapter 1, set the switch to 1 (↑).
5. If you are using a 500 or 601 tape MPC, you also have to set switch 11 to reflect the density of the BCE/Multics tape. If the density is 556 bpi, set the switch to 0 (↓). If the density is 800 bpi, set the switch to 1 (↑). (There is no switch for setting tape density on 610 and 611 MPCs -- they work only with 1600 bpi tapes.)

## Binary Representations

Tape Drive Number	MPC Switch Setting							
	5	6	7	8				
1	0	0	0	1	((↓	(↓	(↓	(↑))
2	0	0	1	0	((↓	(↓	(↑	(↓))
3	0	0	1	1	((↓	(↓	(↑	(↑))
4	0	1	0	0	((↓	(↑	(↓	(↓))
5	0	1	0	1	((↓	(↑	(↓	(↑))
6	0	1	1	0	((↓	(↑	(↑	(↓))
7	0	1	1	1	((↓	(↑	(↑	(↑))
8	1	0	0	0	((↑	(↓	(↓	(↓))
9	1	0	0	1	((↑	(↓	(↓	(↑))
10	1	0	1	0	((↑	(↓	(↑	(↓))
11	1	0	1	1	((↑	(↓	(↑	(↑))
12	1	1	0	0	((↑	(↑	(↓	(↓))
13	1	1	0	1	((↑	(↑	(↓	(↑))
14	1	1	1	0	((↑	(↑	(↑	(↓))
15	1	1	1	1	((↑	(↑	(↑	(↑))

- Go to the bootload IOM. Make sure that the SOURCE (L68)/BOOT SOURCE (DPS 8) switch on the bootload panel is set to TAPE. Also make sure that the CHANNEL NUMBER CODE (L68)/CHANNEL SELECT (DPS 8) switches on the bootload panel reflect the IOM channel number of the tape MPC which controls the tape drive on which you have mounted the BCE/Multics tape. Finally, make sure that the ZERO-BASE S.C. PORT NO (L68)/SCU PORT NUMBER (DPS 8) switches on the bootload panel reflect the number of the SCU port to which the IOM is connected. For an IMU, this information must be verified by examining the configuration file in the IMU.

7. Make sure that the BCE/Multics tape is at the beginning; i.e., that the BOT light on the tape drive is on. If the light is off and the tape is mounted on a 500 tape drive, rewind the tape by pressing the STANDBY button, then the REWIND button, then the READY button on the tape drive. If the light is off and the tape is mounted on a 600, 610 or 630 tape drive, rewind the tape by pressing the STOP button, then the REWIND button, then the START button on the tape drive. Continue with step 9.
8. When using an MTU8200, the bootload device is predefined in the IMU configuration file. This can be changed permanently by using the MCA "CONFIG" command, described in the "Information Multiplexer Unit Hardware Operations Manual, Order No. 58010010". It can also be changed temporarily, along with the channel, by specifying the device number and/or channel number when using the MCA "IBOOT" or "BOOT" commands (i.e., "#01>iboot d00", "#01>iboot c20", or "#01>iboot d00 c20").
9. Check to make sure that all disk packs normally used by the system are on their correct drives. Follow your site policy as to which packs these are and on which drives they belong. If any packs are not on their correct drives, put them there. It is especially important that the Root Physical Volume (RPV) and other Root Logical Volume (RLV) packs be where the config deck says they should be. (This information is on the root card.) The system will assume that everything is in the same place as it was the last time the system was up. So as long as nothing has changed, the system will have no problem mounting the volumes.
10. Initialize every processor *except* the one on which you are running (the bootload processor). To initialize a Level 68 processor, leave its PORT ENABLE switches on, and press the INITIALIZE & CLEAR button on the configuration panel at least 4 times. Each time you press the INITIALIZE & CLEAR button on the Level 68 processor, wait about 5 seconds for the CPU to finish before you press it again. To initialize a DPS 8 processor, leave its PORT ENABLE switches on, and press the INIT-CLEAR button on the configuration panel at least 4 times. Each time you press the INIT-CLEAR button on the DPS 8 processor, wait for the CPU self tests to finish before you press it again. When the self tests are finished, the green light goes on. This light is located on the right hand side of the MP board (the topmost board in the CPU cabinet), just behind the free-edge connector for that board. Since the light is inside the CPU in an awkward position, it may be easier for you to simply wait 10 to 15 seconds before you press the INIT-CLEAR button again.
11. If you are booting from an IOM, continue with step 12. If you are booting from an IMU, continue with step 16.
12. If the buttons on your bootload console are wired to the INITIALIZE and BOOTLOAD functions in the IOM, continue with step 13 (for a 6601) or step 14 (for a 6001/6004). Otherwise, continue with step 15.
13. If the bootload console is a 6601 and you have a system indicator panel next to the console, do the following:
  - Press the INITIALIZE button (located on the panel)
  - Wait for the DATA SET READY light to blink
  - Press the RETURN key



- Wait for the system to respond with:

CONSOLE READY...

- Press the BOOTLOAD button (located on the panel).

If you do not have a system indicator panel at your site, do the following:

- Type ESC CTL-I (press the ESC key, then hold down the CTL key while you press the I key)
- Wait for the DATA SET READY light to blink
- Press the RETURN key
- Wait for the system to respond with:

CONSOLE READY...

- Type ESC CTL-B (press the ESC key, then hold down the CTL key while you press the B key).

Then continue with step 17.

14. If the bootload console is a 6001/6004, do the following:

- Press the INITIALIZE button (located on the console cabinet)
- Press the BOOTLOAD button (located on the console cabinet).

Then continue with step 17.

15. Go to the bootload IOM. Press the SYSTEM INITIALIZE (L68)/SYSTEM INIT (DPS 8) button on the bootload panel. Then press the BOOTLOAD button, also on the bootload panel. Then continue with step 17.

16. There are three ways to boot BCE using an IMU. The examples below assume that the bootload IMU is port 1 and the MCA number is 01. You do not actually type the ">"; the MCA prints it.

- A. If the system indicator panel buttons are wired to the INITIALIZE and BOOTLOAD functions in the IMU, do the following:

- Press the INITIALIZE button (located on the panel)
- Wait for the system to respond with:

```
#----- Console Self-Test in Execution -----#
#----- Control Terminal Display Check, Completed. ----#
```

<displayed on the VIP only>

```
# CONSOLE SELF TEST SUCCESSFUL #
# COPYRIGHT 1983,84 (C) HONEYWELL INFORMATION SYSTEMS #
# IPC CONSOLE READY (F/W TAB 008) #
```

```
# MULTIDROP ENABLED #
```

```
#01<
```

```
#01<MSG MCA IMU INITIALIZE STARTED CONFIG FILE: .CXXXX
```

```
#01<
```

```
#01<MSG MCA INITIALIZATION COMPLETE: NO ERROR
```

```
#01<
```

```
#01<
```

```
#01<Monday September 23, 1985. 09/23/85 (03:11:30)
```

<displayed on both the VIP and the hard copy printer>

- Press the BOOTLOAD button (located on the panel)
- Wait for the system to respond with:

```
#01<
```

```
#01<MSG MCA SYSTEM BOOT IN PROGRESS
```

```
#01<
```

Then continue with step 17.

B. If MCA input is enabled (unlocked), do the following:

- Type:

```
#01<iboot
```

- Wait for the system to respond with:

```
#----- Console Self-Test in Execution -----#
```

```
#----- Control Terminal Display Check, Completed. ----#
```

<displayed on the VIP only>

```
# CONSOLE SELF TEST SUCCESSFUL #
# COPYRIGHT 1983,84 (C) HONEYWELL INFORMATION SYSTEMS #
# IPC CONSOLE READY (F/W TAB 008) #
```

```
# MULTIDROP ENABLED #
```

```
#01<
```

```
#01<MSG MCA IMU INITIALIZE STARTED CONFIG FILE: .CXXXX
```

```
#01<
```

```
#01<MSG MCA INITIALIZATION COMPLETE: NO ERROR
```

```
#01<
```

```
#01<
```

```
#01<Monday September 23, 1985. 09/23/85 (03:11:30)
```

```
#01<
```

<displayed on both the VIP and the hard copy printer>

Then continue with step 17.

C. If MCA input is disabled (locked), do the following:

- Type ESC CTL-I (press the ESC key, then hold down the CTL key while you press the I key)
- Wait for the system to respond with:

```
#----- Console Self-Test in Execution -----#  
#----- Control Terminal Display Check, Completed. -----#
```

<displayed on the VIP only>

```
# CONSOLE SELF TEST SUCCESSFUL #  
# COPYRIGHT 1983,84 (C) HONEYWELL INFORMATION SYSTEMS #  
# IPC CONSOLE READY (F/W TAB 008) #  
  
# MULTIDROP ENABLED #  
#01<  
#01<MSG MCA IMU INITIALIZE STARTED CONFIG FILE: .CXXXX  
#01<  
#01<MSG MCA INITIALIZATION COMPLETE: NO ERROR  
#01<  
#01<  
#01<Monday September 23, 1985. 09/23/85 (03:11:30)
```

<displayed on both the VIP and the hard copy printer>

- Type ESC CTL-B (press the ESC key, then hold down the CTL key while you press the B key)
- Wait for the system to respond with:

```
#01<  
#01<MSG MCA SYSTEM BOOT IN PROGRESS  
#01<
```

Then continue with step 17.

17. The BCE/Multics tape will move, and if the tape drive has a BOT light, it will go out. In addition, the audible alarm on the bootload console will sound. Press ATTN/RESET (6001/6004) or any key (6601) to turn it off.

18. The system will respond on the bootload console with:

```
Booting System MR12.0 generated 07/31/84 1552.5 est Tue.
```

19. If the bootload tape subsystem is controlled by an MPC, which does not have firmware loaded, you must load firmware into the MPC.

The system responds with:

*Enter boot tape MPC model:*

Continue with step 21.

20. If the bootload tape subsystem is not controlled by an MPC, e.g., FIPS tape subsystem, or is controlled by an MPC with operational firmware, then continue with step 23.

### Loading Firmware into the Tape MPC Used to Boot BCE

If a storage device like tape or disk is controlled by an MPC, then before that device can be used, firmware must be loaded into the MPC for that device.

21. At the bootload console, type:

<MPC designation>

for example:

t610

where "t610" designates an MTP0610 tape controller. (If you type "?", the system will give you a list of the valid MPC designations.) Remember that you must press the EOM (6001/6004) or the RETURN (6601) key to enter your response.

22. The system will respond with:

Booting t610 on IOM a chn 14 with m610 rev.11 firmware.  
Booted tape MPC.  
<various status messages>

Continue with step 23.

23. The system will eventually respond with: \*

*Enter rpv data:*

Continue with step 24.

## Entering the Location of the Root Physical Volume (RPV)

*Note:* in the next few steps, the disk controller in question is the one for the bootload disk subsystem (i.e., the subsystem which contains the disk pack that houses the RPV).

24. If the disk pack that houses the RPV is a MSU3380 or a MSU3381, continue with step 25. Otherwise, continue with step 26.
25. At the bootload console, type:

```
rpv <IMU designation> ipc <disk drive model>  
    <disk drive number and subvolume>
```

for example:

```
rpv a20 ipc 3381 0a
```

where "a" is the tag of the IMU in which the disk IPC is located, "20" is the number of the IMU IPC channel, "ipc" is the designation for the IPC-FIPS disk channel in the IMU for these disk types, "3381" is the model of the disk drive, and "0a" indicates that the RPV is located on disk drive "0" and that subvolume "a" of the disk pack is the RPV physical volume.

Then continue with step 27.

26. At the bootload console, type:

```
rpv <IOM designation> <MPC model>  
    <disk drive model> <disk drive number>
```

for example:

```
rpv a24 609 451 1
```

where "a" is the tag of the IOM to which the MPC is connected, "24" is the number of the IOM channel to which the MPC is connected, "609" is the model number of the MPC, "451" is the model number of the disk drive on which the RPV is mounted, and "1" is the number of the disk drive on which the RPV is mounted. (If you type "help", the system will provide you with some information. If you type "?", the system will prompt you for each item one at a time.)

27. The system will respond with:

```
bce (early) 1552.5:
```

## Booting BCE

28. At the bootload console, type:

```
bce
```

29. The system will respond with:

```
The current system time is Tuesday, July 31, 1984
15:52:05 est. Is this correct?
```

### Setting the Calendar Clock

The calendar clock must be set correctly, to avoid causing serious damage to the storage system. *Note:* the calendar clock is always set to the local time. Daylight savings time is specified via the klok card in the configuration deck.

30. Check the time zone in the message. Your check should be especially careful if you have recently switched to daylight savings time or vice versa. If the time zone is not correct, type:

```
abort
```

then continue with step 31. If the time zone is correct, check the time in the message. It will almost always be wrong, so you should type:

```
no
```

then continue with step 39.

### EDITING THE CLOK CARD

31. At the bootload console, enter the config deck editor by typing:

```
config
```

32. Print the klok card by typing:

```
/klok/
```

33. The system will respond with:

```
klok -delta +05. -zone est -boot_delta 12.
```

where "+05." is the number of hours (in decimal) that local time is earlier than Greenwich mean time, "est" is the time zone, and "12." is the maximum number of hours (in decimal) you ever expect to have Multics down. This number should reflect your site's normal interval between shutdowns and startups. Having this number on the klok card helps protect against bad clock settings.

34. Let your system administrator know that you are changing the klok card. Then type:

```
s/<bad time difference>/<good time difference>/
s/<bad time zone>/<good time zone>/p
```

for example:

```
s/+05./+04./  
s/est/edt/p
```

where "+04." is the correct number of hours (in decimal) that local time is earlier than Greenwich mean time, and "edt" (Eastern Day Light Time) is the correct time zone.

35. The system will respond with:

```
clock -delta +04. -zone edt -boot_delta 12.
```

36. Make the change to the clock card permanent by typing:

```
w
```

37. Get out of the config deck editor by typing:

```
q
```

38. The system will respond with:

```
bce (early) 1559.6:
```

Now start over with step 28.

#### SETTING THE TIME

39. The system will respond with:

```
Enter time:
```

40. Type:

```
<month>/<day>/<year> <hour>:<minutes>
```

for example:

```
07/31/84 15:58
```

Choose a time which is a minute or so ahead of the current time, to give yourself time to perform the next step.

41. The system will respond with:

```
The current system time is Tuesday, July 31, 1984 15:58:02  
est. Is this correct?
```

42. Type:

yes

then, at the instant when the current time reaches the time you typed, press the EOM (6001/6004) or the RETURN (6601) key.

43. The system will respond with:

```
load_disk_mpcs: Disk mpcs mspa mspb mspc appear not
to be operating. Enter disk mpc names to be loaded,
or "none" or "abort" or "all":
```

#### Loading Firmware into the Disk MPCs

44. Look at the disk MPC names displayed in the message. Usually, all of the disk MPCs whose names are in the message will need to have firmware loaded into them. In this case, type:

all

If some of the disk MPCs should not have firmware loaded into them, just type the names of those which should have firmware loaded into them. Type the names as they appear in the message. For example:

mspa mspb

If there are problems with any of the names displayed in the message (for example, one of them is "mspg" and you do not have an mspg), type:

abort

then ask your system maintainer for help. Do not continue with this procedure.

45. Assuming you did not type "abort," the system will respond with:

```
bce (boot) 1601.3:
```

#### Loading Firmware into the Other MPCs

46. Load firmware into any remaining tape MPCs and unit record controllers by typing:

```
fwload <controller names>
```

for example:

```
fwload mtpa urpa
```



47. The system will respond with:

```
Booting mtpa on IOM a chn 22...
Booting urpa on IOM a chn 24...
bce (boot) 1602.7:
```

### Resynchronizing the Paper in the Line Printers

After you load all of the firmware, you will have to resynchronize the paper in each line printer.

48. Open the printer door.
49. Declutch the tractors. On a PRU0901/1201 or PRU0903/1203 printer, this means moving the orange switch to the left of the paper (marked PUSH TO CLUTCH/TRACTORS/PUSH TO DECLUTCH) to the correct position. On a PRU1200/1600 printer, this means pushing down the white lever on the left side of the bar behind the paper until the paper can turn.
50. Line up a paper fold that points inward (toward the printer) with the red line at the top of the right hand tractor, by hand-rotating the bar behind the paper.
51. Hit the SKIP button.
52. Reclutch the tractors. On a PRU0901/1201 or PRU0903/1203 printer, this means moving the orange switch to the correct position. On a PRU1200/1600 printer, this means raising the white lever as far as it will go.
53. Hit the SKIP button again, then the START button.
54. Repeat this procedure for each remaining line printer.

## PROBLEMS

### Initializing BCE

*Note:* before you repeat any part of the initialization procedure, you should make sure that the BCE/Multics tape is at the beginning: i.e., that the BOT light is on. If the light is off and the tape is mounted on a 500 tape drive, rewind the tape by pressing the STANDBY button, then the REWIND button, then the READY button on the tape drive. If the light is off and the tape is mounted on a 600, 610 or 630 tape drive, rewind the tape by pressing the STOP button, then the REWIND button, then the START button on the tape drive. If the tape is mounted on an 8205, 8206, or 8208 tape drive, rewind the tape by pressing the RESET button, then the LOAD/REWIND button, then the START button on the tape drive.

1. Sometimes the BCE/Multics tape and tape drive do not operate correctly on the first attempt. If nothing happens after you press the console buttons or give the iboot command for the IMU, press the buttons or give the command again. Try the bootload sequence at least three times before trying something else.

2. If nothing happens after you try the bootload sequence several times, and if the tape subsystem is on an MPC, double-check the tape MPC switches. The switches should reflect the BCE/Multics tape drive, MPC link adapter, and tape density. If they do not, start over at step 2. If the tape subsystem is on an IMU, or if the BCE/Multics tape drive is an 8200, verify that the BCE/Multics tape is on the drive defined in the MCA configuration file. If it is not, you can change the default tape drive by adding the "dxx" argument to the MCA iboot command, where "xx" is the number of the new tape drive. To do this, start over with step 15. *Note:* if you use the "dxx" control argument with the iboot command, the tape drive specified by "xx" becomes the default bootload tape drive.
3. If the MPC switches are OK, double check the SOURCE (L68)/BOOT SOURCE (DPS 8) switch on the bootload IOM to make sure it is set to TAPE. If it is not, start over at step 6.
4. If the SOURCE (L68)/BOOT SOURCE (DPS 8) switch is OK or the BCE/Multics tape is on the drive defined in the MCA configuration file, try mounting the BCE/Multics tape on another tape drive, and repeating steps 2 through 17. If your site has multiple copies of the current BCE/Multics tape, you can also try using another tape and repeating steps 1 through 17.
5. If the BCE/Multics tape does not move and the BOT light is still on, go to the SCU and press the INITIALIZE and PANEL CLEAR buttons on the maintenance panel. Then start over at step 10.
6. If the TROUBLE light on either the CPU, the IOM, or the SCU is on, start over at step 10.
7. If all else fails, take a look at the console printer and make sure it has not run out of paper.

#### Loading Firmware into the Tape MPC Used to Boot BCE

8. If you type the wrong tape MPC, the system will respond with the following message:  

```
Unknown tape MPC model 016t
```

If this happens, just type the correct information, and continue with the next step.

9. If you get the following message:

```
Enter tape drive number for memory dump:
```

it either means that you have hardware problems, that some of the switches are not set right, or that the config deck is wrong. Unless you have been instructed to type a number in response to this message, you may ignore it. Check all of the switches (refer to Section 9 and Appendix A). If some of them are wrong, correct them and start over with step 1. If the switches are right, check the config deck (refer to Section 14). If it is wrong, correct it and start over with step 1. If it is right, ask your system maintainer for help.

#### Booting BCE

10. If any other messages are printed, check to see that the OPERATING MODE switch on the IOM bootload panel is set to PAGED. Also check to see that the OPERATING MODE switch on the CPU configuration panel is set to MULTICS (L68) or to VMS

(DPS 8). If either OPERATING MODE switch is not set correctly, change the setting and start over with step 5.

11. If the volume label (of the disk drive on which the pack containing the BCE partition is mounted) can not be interpreted, you will get a message like one of the following:

```
init_root_vols: Error <explanation> reading RPV label.
```

```
init_root_vols: dska_01 is not the RPV, but rather is:  
root3.
```

Check to see that the right pack is mounted on the drive. Also check to see that you typed the right IOM channel number and disk drive number in the rpv command. If the wrong pack is mounted or you typed the wrong numbers, start over with step 23. If everything seems to be OK, ask your system maintainer for assistance.

12. If you get the following message:

```
scs_and_clock_init: The zone on the klok card is not in  
time_info_.
```

it means that the time zone on the klok card is not one that the system accepts; i.e., it is not listed in the system's table of time zones. (For a list of the acceptable time zones, see the description of the klok card in the *Multics System Maintenance Procedures* manual, Order No. AM81.)

BCE will crash and you will remain in the "early" state. Edit the klok card (see steps 30-37). Then reboot BCE by starting over with step 27.

### Setting the Calendar Clock

13. If you set the calendar clock to a time which is completely unreasonable, the system prints this message:

```
This is clearly incorrect.
```

and asks you to reenter the time.

14. If you set the calendar clock to a time which is before the last shutdown time, the system prints this message:

```
The current time is before the unmounted time in the RPV  
label. Is this correct?
```

If you set the calendar clock to a time which is more than the normal number of hours after the last shutdown time, the system prints this message:

```
The current time is more than the supplied boot_delta  
hours beyond the unmounted time in the RPV label.  
Is this correct?
```

("boot\_delta" is a field on the klok card.) In each case, you should answer the question, and if necessary, reenter the time.

### **Loading Firmware into the Disk and Other MPCs**

15. If you are not sure of what name to use for one of your MPCs, type:

help

The system will respond with a list of valid controller names.

16. If the system fails, start over with step 1.
17. If the system fails repeatedly trying to load firmware into a particular MPC, make sure the MPC is powered on. If it is not, power it on and try loading firmware into it again. If it is powered on, check to see that switch 2 or 10 is set correctly. (MPC switch 2 or 10 should reflect the number of the MPC link adapter that is connected to the IOM through which you are loading the MPC firmware. See step 4 for more details.) If the switch is set correctly, the MPC is probably broken, and you should notify CSD.
18. If the HALTED light on an MPC configuration panel lights up, the firmware load for that MPC is bad. Push the INITIALIZE button, then the START button on the MPC config panel. Then load the firmware for that MPC again.

\*



## SECTION 12

# EDITING THE CONFIG DECK TO CHANGE HARDWARE STATES

Usually, booting BCE is all you have to do to tell the software how it should \*  
use the hardware: what hardware is available, how the software should set itself up on  
that hardware, and what state the hardware is in.

But sometimes, your system administrator or maintainer will ask you to change  
the state of a major hardware module or a peripheral. Changing the state of a box  
or a peripheral means making it *active* (turning it on if it is off) or making it  
*inactive* (turning it off if it is on).

To change the state of a box or a peripheral, you must edit cards in the  
config deck. Editing config cards can *ONLY* be done at BCE level. \*

Config cards may be in either standard format or labeled format. In labeled  
format, each field on a config card, except for the card name, is preceded by a label.  
Thus, the following card in standard format:

```
cpu a 3 on dps8 70.
```

looks like this in labeled format:

```
cpu -tag a -port 3 -state on -type dps8 -model 70.
```

This section explains the procedures for changing the state of a processor, a  
system controller, an IOM, an FNP, and a system console. \*

\*

### CHANGING THE STATE OF A PROCESSOR

To change the state of a processor, you must edit its cpu card.

1. At the bootload console, enter the config deck editor by typing:

```
config
```

2. Print the first cpu card in the deck by typing:

```
/cpu/
```

3. The system will respond with something like:

```
cpu -tag a -port 3 -state on -type dps8 -model 70.
```

where "a" is the tag of the CPU, "3" is the number of the SCU port to which it is connected, "dps8" is the type of CPU it is, and "70." is its model number.

4. If this is the card you want to edit, continue with step 5 or step 7. If it isn't, repeat step 2 until you get the card you want to edit. Then continue with step 5 or step 7.
5. To make CPU b an active processor, type:

```
s/off/on/p
```

6. The system will respond with:

```
cpu -tag b -port 4 -state on -type 168 -model 80.
```

7. To make CPU c an inactive processor, type:

```
s/on/off/p
```

8. The system will respond with:

```
cpu -tag c -port 7 -state off -type dps8 -model 70.
```

9. Make the change to the cpu card permanent by typing:

```
w
```

10. Get out of the config deck editor by typing:

```
q
```

11. The system will respond with:

```
bce (boot) 0836.2:
```

## CHANGING THE STATE OF A SYSTEM CONTROLLER

To change the state of a system controller, you must edit its mem card.

1. At the bootload console, enter the config deck editor by typing:

```
config
```

2. Print the first mem card in the deck by typing:

```
/mem/
```

3. The system will respond with something like:

```
mem -tag a -size 1024. -state on
```

where "a" is the tag of the SCU and "1024." specifies the amount of memory in the SCU.

4. If this is the card you want to edit, continue with step 5 or step 7. (*Note:* the bootload SCU is the first SCU listed in the deck, and must be "on.") If it isn't, repeat step 2 until you get the card you want to edit. Then continue with step 5 or step 7.

5. To make SCU b an active system controller, type:

```
s/off/on/p
```

6. The system will respond with:

```
mem -tag b -size 1024. -state on
```

7. To make SCU c an inactive system controller, type:

```
s/on/off/p
```

8. The system will respond with:

```
mem -tag c -size 1024. -state off
```

9. Make the change to the mem card permanent by typing:

```
w
```

10. Get out of the config deck editor by typing:

```
q
```

11. The system will respond with:

```
bce (boot) 0836.2:
```

## CHANGING THE STATE OF AN IOM

To change the state of an IOM, you must edit its iom card.

1. At the bootload console, enter the config deck editor by typing:

```
config
```

2. Print the first iom card in the deck by typing:

```
/ iom/
```



3. The system will respond with something like:

```
iom -tag a -port 0 -model iom -state on
```

where "a" is the tag of the IOM, "0" is the number of the SCU port to which it is connected, and "iom" specifies what model it is.

4. If this is the card you want to edit, continue with step 5 or step 7. If it isn't, repeat step 2 until you get the card you want to edit. Then continue with step 5 or step 7.

5. To make IOM b an active IOM, type:

```
s/off/on/p
```

6. The system will respond with:

```
iom -tag b -port 1 -model iom -state on
```

7. To make IOM a an inactive IOM, type:

```
s/on/off/p
```

8. The system will respond with:

```
iom -tag a -port 0 -model iom -state off
```

9. Make the change to the iom card permanent by typing:

```
w
```

10. Get out of the config deck editor by typing:

```
q
```

11. The system will respond with:

```
bce (boot) 0836.2:
```

## CHANGING THE STATE OF AN FNP

To change the state of an FNP, you must edit its prph card.

1. At the bootload console, enter the config deck editor by typing:

```
config
```

2. Print the first prph fnp card in the deck by typing:

```
/fnp/
```

3. The system will respond with something like:

```
prph -device fnpa -iom a -chn 28. -model 6670. -state on
```

where "fnpa" indicates which FNP is being described, "a" is the tag of the IOM to which the FNP is connected, "28." is the number of the IOM port to which it is connected, "6670." is the model number of the FNP, and "on" is the state of the FNP.

4. If this is the card you want to edit, continue with step 5 or step 7. If it isn't, repeat step 2 until you get the card you want to edit. Then continue with step 5 or step 7.

5. To make FNP b an active FNP, type:

```
s/off/on/p
```

6. The system will respond with:

```
prph -device fnpb -iom a -chn 29. -model 6670. -state on
```

7. To make FNP a an inactive FNP, type:

```
s/on/off/p
```

8. The system will respond with:

```
prph -device fnpa -iom a -chn 28. -model 6670. -state off
```

9. Make the change to the prph card permanent by typing:

```
w
```

10. Get out of the config deck editor by typing:

```
q
```

11. The system will respond with:

```
bce (boot) 0836.2:
```

## CHANGING THE STATE OF A SYSTEM CONSOLE

To change the state of a system console, you must edit its prph card.

1. At the bootload console, enter the config deck editor by typing:

```
config
```

2. Print the first prph opc card in the deck by typing:

```
/opc/
```

3. The system will respond with something like:

```
prph -device opca -iom a -chn 20. -model 6004. -ll 80.  
-state on
```

where "opca" indicates which system console is being described, "a" is the tag of the IOM to which the console is connected, "20." is the number of the IOM port to which it is connected, "6004." is the model number of the console, "80." is the line length of the console, and "on" is the state of the console.

4. If this is the card you want to edit, continue with step 5 or step 7. If it isn't, repeat step 2 until you get the card you want to edit. Then continue with step 5 or step 7.
5. To make console b (opcb) an active console (i.e., the bootload console), type:

```
s/off/on/p
```

6. The system will respond with:

```
prph -device opcb -iom a -chn 24. -model 6601. -ll 80.  
-state on
```

7. To make console a (opca) an inactive console, type:

```
s/on/off/p
```

8. The system will respond with:

```
prph -device opca -iom a -chn 20. -model 6004. -ll 80.  
-state io
```

9. Make the change to the prph card permanent by typing:

```
w
```

10. Get out of the config deck editor by typing:

```
q
```

11. The system will respond with:

```
bce (boot) 0836.2:
```

## NOTES

1. If you want to change the state of more than one box at a time, you can just enter the config deck editor once, make all your changes, write them, and get out of the editor. You don't have to get into and out of the editor for each change.

## SECTION 13

# BOOTLOADING MULTICS

Multics must be booted when the system has been shut down, when there has been a system failure which has caused the system to return to BCE, and when a new version of Multics needs to be installed.

Under normal circumstances, you will boot Multics in one of two ways: with the *boot command* or with the *auto exec\_com*.

The *boot command* boots Multics from the information on the RPV and may also initialize the answering service.

The *auto exec\_com* does what the *boot command* does, and also puts the system in automatic mode. (Automatic mode is explained in Section 31.)

\*

### TO BOOT MULTICS FROM BCE

1. At the bootload console, type:

```
ec auto star
```

if your site uses the *auto exec\_com*, or:

```
boot star
```

if your site uses the *boot command*, where "star" is the short form of the startup command.

2. If you typed "ec auto star," the system will respond by printing the cpu, mem and iom cards from the config deck, then with various status messages.
3. After several minutes, the system will respond with:

```
Multics 12.0 - 07/31/84 0845.1 est Tue
```

This message tells you that the initializer process is ready to go. Check the date and time to be sure both are correct. If the time is wrong, you must shut down and correct the clock. See Section 17 "Shutting the System Down," and Section 11.

4. After about 90 seconds, the system will respond with messages *like* the following:

```

Loading FNP B, >scl>mcs 6.7
FNP B loaded successfully
as Multics 12.0; Answering Service 16.0
as LOGIN Data_Management.Daemon dmn dml (create)
as LOGIN 10.SysDaemon dmn cord (create)
as LOGIN Backup.SysDaemon dmn bk (create)
as LOGIN 10.SysDaemon dmn prta (create)
as LOGIN Volume_Dumper.Daemon dmn vinc (create)
dml Initializing Data Management in ring 2.
dml Data Management System initialized.
dml Initializing Data Management in ring 2.
dml Data Management System initialized.
cord Enter command: coordinator or driver
Loading FNP A, >scl>mcs 6.7
FNP A loaded successfully
prta Enter command: coordinator or driver
->cord
->prta
vinc r 852 1.083 21.705 348
vinc
->vinc
bk r 852 2.267 34.136 503
bk
->bk
Ready (Not signed on.)

```

5. These messages tell you that the answering service is being initialized, the FNPs are being loaded, the daemons are logging in, and that automatic system routines are running.
6. Remember that if your site requires operator authentication, you must sign on on the bootload console and any initializer terminals you plan to use before you can enter commands. See "Operator Authentication" in Section 4.

## PROBLEMS

1. If an FNP doesn't load, you'll get some error messages, including:

```
FNP A not loaded
```

Type:

```
load_mpx <FNP tag>
```

for example:

```
load_mpx a
```

where "a" is the tag of the FNP mentioned in the messages. If this succeeds, the system will respond with:

```
FNP A loaded successfully
```

If it doesn't, ask your system maintainer for assistance.

2. If you get the following message: \*

```
scs_and_clear_init: The zone on the clock card is not in
time_info_.
```

it means that the time zone on the clock card is not one that the system accepts; i.e., it is not listed in the system's table of time zones. (For a list of the acceptable time zones, see the description of the clock card in the *Multics System Maintenance Procedures* manual, Order No. AM81.) BCE will crash and you will go to the "bce\_crash" state. Edit the clock card (see steps 23-30 of "To Boot BCE from Scratch" in Section 11). Then reboot Multics by starting over with step 1.

3. If you try to boot Multics with a nonroot volume where a RLV should be, the system will fail. You'll get one of the following error messages:

```
init_pvt: no root
init_pvt: dska 3 has no Multics label
init_pvt: no partition hc on root dska 3
```

Mount the correct packs on the correct drives. Then start over at step 1.

4. If a logical volume can't be accepted by the system, you've either mounted a wrong pack, or you've mounted a right pack on a wrong drive. You'll get an error message, and the system will wait at ring 1 command level. Mount the correct pack on the correct drive. Then type:

```
star
```

where "star" is the short form of the startup command.

If this doesn't work, type:

```
dlv <logical volume>
```

for example:

```
dlv Public
```

where "dlv" is the short form of the del\_lv command, and "Public" is the name of the logical volume mentioned in the error message. *Note:* if the logical volume mentioned in the error message is the Root Logical Volume, the system will not let you delete it. You must make the RLV available to the system, because Multics can't run without it.) Then type:

```
star
```

The system will continue booting Multics. If you have to use the dlv command, let your system administrator know, so he or she can determine the impact on users and process directory volumes.

5. If the system crashes during startup, it will return to BCE. You should do the following:

- First, dump Multics and do an emergency shutdown.
- Second, read the error messages and correct the problem.
- Third, start over at step 1.

If the system crashes again, dump Multics and do an emergency shutdown, then ask your system maintainer for assistance. If you can't take the second dump without overwriting the first dump, ask your system administrator which dump he or she wants. Details on crashes are given in Sections 35 and 36. See also "Dumping Multics" and "Performing ESD" in Section 37.

6. If the answering service fails during startup, the system will remain in Multics. You should do the following:

- First, if the error messages give you enough information to figure out what's wrong, correct the problem.
- Second, shut the system down.
- Third, start over at step 1.

If this doesn't work, or if you can't figure out what's wrong from the error messages, ask your system maintainer for assistance. See Section 17, "Shutting the System Down."

7. If the system starts up, but seems to be running slowly, the CACHE ENABLE switches inside the processor may be set incorrectly. Ask your system maintainer to reset them.

## SECTION 14

# LOGGING IN THE DAEMONS -- I/O

The I/O daemons usually get logged in automatically by the `system_start_up.ec`, which runs when you boot Multics. All you have to do is start them. When you start an I/O daemon, the driver process is initialized and the daemon begins operating.

If the I/O daemons don't get logged in automatically at your site, or if they get logged out due to problems, you'll have to log them in yourself, then start them. Your site may have its own `x` command(s) for logging in and starting the particular I/O daemons in use at your site. *Note:* you should log in the coordinator before you log in any of the drivers.

If your site uses AIM, you may have to specify a *device class* in addition to a *request type* when you start up a driver. A request type is a set of queues into which users may place requests for the I/O daemon to perform services. A device class is a subdivision of a request type by access class.

### TO START THE I/O DAEMONS

1. Type:

```
x io -all
```

to start all of the I/O daemons, or:

```
x iol
```

to start the coordinator and the minimal set of drivers defined by your site, or:

```
x io
```

to start the coordinator and printer driver.



2. The system will respond with messages like the following:

```
cord I/O Coordinator version: 3.1
cord I/O Coordinator initialized
->cord
prta Enter device name and optional request type:
cord New driver for device prta, request type printer
(series = 10000)
RCP: Attached prta for 10.SysDaemon.z
as Check separator bar alignment
as reply go to prta when it is ok; reply sample_hs to
see new alignment
prta
prta Parameters set for the standard printer request type.
prta
prta prta driver ready at 06/01/83 0852.5 est Wed
prta
prta Enter command:
```

3. The audible alarm on the bootload console will sound and a sample page will be printed.
4. Check the page. If it's OK, start processing requests by typing:

```
r <printer driver label> go
```

for example:

```
r prta go
```

5. If the page isn't OK, fix the alignment, then type:

```
r prta sample_hs
```

6. The system will respond with:

```
prta Enter command:
```

Another sample page will be printed. Repeat step 5 until the page is OK, then start processing requests by typing:

```
r prta go
```

7. The system will respond with:

```
Ready (MacKenzie)
```

If your site has more than one printer driver, you will need to repeat steps 3 through 7 for each one. Printer drivers which control remote printers start running without the sample\_hs and go commands.

## TO LOG IN AND START THE I/O DAEMON COORDINATOR

You may have to use this procedure if your site doesn't use the "x io" command described earlier.

1. Type:

```
login 10.SysDaemon <l/O daemon coordinator label>
```

for example:

```
login 10.SysDaemon cord
```

If your site uses AIM, you should specify the authorization of the coordinator by typing this instead:

```
login 10.SysDaemon cord -auth system_high
```

2. The system will respond with:

```
as LOGIN 10.SysDaemon dmn cord (create)
cord Enter command: coordinator or driver
->cord
```

3. Type:

```
r cord coordinator
```

4. The system will respond with:

```
cord l/O Coordinator Version: 3.1
cord l/O Coordinator initialized
```

## TO LOG IN AND START A PRINTER I/O DAEMON DRIVER

You may have to use this procedure if your site doesn't use the "x io" command described earlier.

1. Type:

```
login <daemon User_id> <printer driver label>
```

for example:

```
login 10.SysDaemon prta
```

If your site uses AIM, you should specify the authorization of the printer driver by typing this instead:

```
login 10.SysDaemon prta -auth <authorization>
```

Your system administrator will tell you what the authorization should be. (Note: if your site doesn't use IO.SysDaemon, the daemon User\_id may have the right authorization associated with it, making this unnecessary.)

2. The system will respond with:

```
as LOGIN IO.SysDaemon dmn prta (create)
prta Enter command: coordinator or driver
->prta
```

3. Type:

```
r prta driver
```

4. The system will respond with:

```
prta Enter device name and optional request type:
->prta
```

5. If you want the printer to run with the default request type (and the default device class), type:

```
r <printer driver label> <device name>
```

for example:

```
r prta prta
```

(At most sites, the printer driver label and the device name are the same.)

If you want the printer to run with a request type (or a device class) other than the default, type:

```
r <printer driver label> <device name> <device request type>
```

for example:

```
r prta prta unlined
```

or, if your site uses device classes:

```
r <printer driver label> <device name>
  <device request type>.<device class>
```

for example:

```
r prta prta unlined.confidential
```

6. The system will respond with:

```
RCP: Attached prta for IO.SysDaemon.z
prta prta driver ready at 06/01/83 0855.1 est Wed
->prta
```

7. To start processing requests, type:

```
r prta go
```

8. The system will respond with:

```
Ready (MacKenzie)
```

## TO LOG IN AND START A PUNCH I/O DAEMON DRIVER

You may have to use this procedure if your site doesn't use the "x io" command described earlier.

1. Type:

```
login <daemon User_id> <punch driver label>
```

for example:

```
login IO.SysDaemon puna
```

If your site uses AIM, you should specify the authorization of the punch driver by typing this instead:

```
login IO.SysDaemon puna -auth <authorization>
```

Your system administrator will tell you what the authorization should be. (Note: if your site doesn't use IO.SysDaemon, the daemon User\_id may have the right authorization associated with it, making this unnecessary.)

2. The system will respond with:

```
as LOGIN IO.SysDaemon dmn puna (create)
puna Enter command: coordinator or driver
->puna *
```

3. Type:

```
r puna driver
```

4. The system will respond with:

```
puna Enter device name and optional request type:
->puna
```

5. If you want the punch to run with the default request type (and the default device class), type:

```
r <punch driver label> <device name>
```

for example:

```
r puna puna
```

(At most sites, the punch driver label and the device name are the same.)

If you want the punch to run with a request type (or a device class) other than the default, type:

```
r <punch driver label> <device name> <device request type>
```

for example:

```
r puna puna punch2
```

or, if your site uses device classes:

```
r <punch driver label> <device name>  
  <device request type>.<device class>
```

for example:

```
r puna puna punch2.confidential
```

6. The system will respond with:

```
RCP: Attached puna for 10.SysDaemon.z  
puna puna driver ready at 06/01/83 0855.1 est Wed  
->puna
```

7. To start processing requests, type:

```
r puna go
```

8. The system will respond with:

```
Ready (MacKenzie)
```

\*

## TO GET INFORMATION ABOUT DEVICES AND REQUEST TYPES (AND DEVICE CLASSES)

1. If you aren't sure what devices and request types (and device classes, if your site uses them) are available, you can list them by typing:

```
r <I/O daemon coordinator label> print_devices
```

for example:

```
r cord print_devices
```

2. The system will respond with something like the following:

Device	Request type	Access name	
* prta	printer	10.SysDaemon	system_high
* prtb			
* prtc			
* prtd			
prta	test	10.SysDaemon	system_high
prtb			
prtc			
prtd			
* puna	punch	10.SysDaemon	system_high
* punc			
prta	unlined	10.SysDaemon	system_high
prtb			
prtc			
prtd			
* rdra	reader_dummy	10.SysDaemon	
* rdrb			
* rdrc			
* rdrd			

The devices are grouped according to the request types they service. An asterisk (\*) before a device indicates that the associated request type is the default for the device. The driver access name and authorization (if any) are given for each request type.

3. To find out what devices and request types (and device classes, if your site uses them) are already running, type:

```
r <l/O daemon coordinator label> list
```

for example:

```
r cord list
```

4. The system will respond with something like the following:

```
device prta is active, request type printer, request 40017
device prtb is active, request type unlined.confidential,
request 20007
device rem1 is active, request type rem1_print_unlined,
request 50002
device rem2.elite is active, request type printer, request 70003
device rem2.pica is active, request type rem2_pica_unlined,
request 30015
```

## TO FIND OUT WHICH I/O DAEMONS ARE LOGGED IN

1. Type:

```
who <daemon User_id>
```

for example:

```
who IO.SysDaemon
```

2. The system will respond with a message like this:

```
06/01/83  
0810.0 cord 1.0 IO D IO.SysDaemon  
  
0832.3 prta 1.0 IO D IO.SysDaemon
```

## PROBLEMS

1. If the coordinator has already been logged in, the system will print this error message:

```
iod_overseer_: Coordinator is already running.
```

This is not a fatal error. You can ignore it.

2. The first thing the coordinator tries to do is initialize itself. The first step of this initialization is to finish any work left undone by the last coordinator process. If this step can't be performed, the system will print this error message:

```
iodc_init: Warning--Cannot get old saved list. Some deletions  
may not be performed.
```

This is not a fatal error. You can ignore it.

3. If the coordinator encounters any other problems, the coordinator prints this error message:

```
Process cannot be initialized.
```

This is a fatal error and you should bring it to the attention of your system maintainer.

4. If you try to log in a driver when you haven't logged in the coordinator, the driver waits for you to log in the coordinator. If you don't log in the coordinator within five minutes, the driver sends you an error message and logs itself out.

5. The first thing a driver tries to do is initialize itself. If it encounters a problem which isn't serious, the system will respond with an error message and the driver will ask you for instructions. If it encounters a problem which is serious, the system will respond with an error message that starts with the words "Fatal error:" and the driver may log out. If you get a fatal error like this, you should bring it to the attention of your system maintainer.
6. If there's some reason why the driver is not allowed to operate the device(s) or the request type(s) (or the device class(es), if your site uses them) you specify, the system will respond with one or more error messages explaining the problem, then ask you again for a command or a device name and optional request type.





## SECTION 15

# LOGGING IN THE DAEMONS -- INCREMENTAL BACKUP

The incremental backup daemons usually get logged in automatically by the `system_start_up.ec`, which runs when you boot Multics. All you have to do is start them. Of course, if your site only runs one of the incremental dumpers, then you only have to start one daemon.

If the incremental backup daemons don't get logged in automatically at your site, or if they get logged out due to problems, you'll have to log them in yourself, then start them.

### TO START THE HIERARCHY INCREMENTAL DUMPER

1. Type:

```
x inc <your initials> <tape name>
```

for example:

```
x inc klm ih001
```

where "ih001" is the name of the first tape to be used.

2. The system will respond with messages like the following:

```
sc_command r bk start_dump sys_dirs klm 1 60
sc_command r bk ih001
bk r 0941 0.324 19.431 152
RCP: Attached tapa_01 for Backup.SysDaemon.z
RCP: Note (tapa_01) -ih001,sys
RCP: Unloading volume from device tapa_01
RCP: Mount Reel ih001 with ring on tapa_01 for
      Backup.SysDaemon.z
bk
->bk
```

3. Mount the specified tape on the specified drive. (See "To Mount a Tape" in Section 19.)

## TO START THE VOLUME INCREMENTAL DUMPER

1. Type:

```
x vinc <your initials>
```

for example:

```
x vinc klm
```

where "vinc" is the short form of the incremental\_volume command.

2. The system will respond with:

```
vinc  
->vinc  
vinc Mounting tape iv001 for writing  
RCP: Mount Reel iv001 with ring on tapa_01 for  
Volume_Dumper.Daemon
```

3. Mount the specified tape on the specified drive. (See "To Mount a Tape" in Section 19.)

## TO LOG IN THE HIERARCHY INCREMENTAL DUMPER

1. Type:

```
login Backup.SysDaemon <backup daemon label>
```

for example:

```
login Backup.SysDaemon bk
```

where "bk" is your site's label for the backup daemon that controls hierarchy incremental and consolidated dumps.

2. The system will respond with:

```
as LOGIN Backup.SysDaemon dmn bk (create)  
bk r 0940 2.267 34.136 503  
bk  
->bk
```

## TO LOG IN THE VOLUME INCREMENTAL DUMPER

1. Type:

```
login Volume_Dumper.Daemon <backup daemon label>
```

for example:

```
login Volume_Dumper.Daemon vinc
```

where "vinc" is your site's label for the backup daemon that controls volume incremental dumps.

2. The system will respond with:

```
as LOGIN Volume_Dumper.Daemon dmn vinc (create)
vinc r 0940 2.134 27.369 678
vinc
->vinc
```

## TO FIND OUT WHICH HIERARCHY BACKUP DAEMONS ARE LOGGED IN

1. Type:

```
who .SysDaemon
```

2. The system will respond with a message like this:

```
06/01/83 0714.4 bk 1.0 bk System D Backup.SysDaemon
```

## TO FIND OUT WHICH VOLUME BACKUP DAEMONS ARE LOGGED IN

1. Type:

```
who .Daemon
```

2. The system will respond with a message like this:

```
06/01/83 2031.2 vinc 1.5 vinc System D Volume_Dumper.Daemon
```



## SECTION 16

# BRINGING THE SYSTEM UP -- THE WHOLE PROCEDURE

Once you're familiar with all of the steps involved in bringing the system up, you can use the script below as a quick reference to the whole procedure. Remember\* that your site may have exec\_coms and x commands of its own for doing parts of this procedure. Don't worry if the time string on some output messages during bootload is inaccurate, as the clock config card still needs to be read.

\*

### BOOTING BCE AND MULTICS

Booting System MR12.0 generated 07/31/84 1552.5 est Tue.  
Enter boot tape MPC model: t610

Booting t610 on IOM a chn 14 with m610 rev.11 firmware.  
Booted tape MPC.

<various status messages>  
Enter rpv data: rpv a24 609 451 1

bce (early) 1552.5: bce

The current system time is Tuesday, July 31, 1984 15:52:05 est.  
Is this correct? no

Enter time: 07/31/84 15:58

The current system time is Tuesday, July 31, 1984 15:58:02 est.  
Is this correct? yes <EOM or RETURN>

load\_disk\_mpcs: Disk mpcs mspa, mspb, mspc appear not to be operating.  
Enter disk mpc names to be loaded, or "none" or "abort" or "all": all

bce (boot) 1601.3: fwload mtpa urpa

Booting mtpa on IOM a chn 22...  
Booting urpa on IOM a chn 24...

bce (boot) 1602.7: ec auto star

Multics 12.0 - 06/01/83 0845.1 est Wed

Loading FNP B, >sci>mcs 6.7  
FNP B loaded successfully  
as Multics 12.0; Answering Service 16.0  
as LOGIN Data\_Management.Daemon dmn dml (create)

```

as LOGIN 10.SysDaemon dmn cord (create)
as LOGIN Backup.SysDaemon dmn bk (create)
as LOGIN 10.SysDaemon dmn prta (create)
as LOGIN Volume_Dumper.Daemon dmn vinc (create)
dm1 Initializing Data Management in ring 2.
dm1 Data Management System initialized.
cord Enter command: coordinator or driver
Loading FNP A, >sc1>mcs 6.7
FNP A loaded successfully
prta Enter command: coordinator or driver
->cord
->prta
vinc r 852 1.083 21.705 348
vinc
->vinc
bk r 852 2.267 34.136 503
bk
->bk
Ready (Not signed on.)

sign_on MacKenzie

Password:

sign_on: MacKenzie signed on as operator on channel otw_.

Ready (MacKenzie)

x io -all

cord I/O Coordinator version: 3.1
cord I/O Coordinator initialized
->cord
prta Enter device name and optional request type:
cord New driver for device prta, request type printer (series = 10000)
RCP: Attached prta for 10.SysDaemon.z
as Check separator bar alignment
as reply go to prta when it is ok; reply sample_hs to see new alignment
prta
prta Parameters set for the standard printer request type.
prta
prta prta driver ready at 06/01/83 0852.5 est Wed
prta
prta Enter command:

r prta sample_hs

prta Enter command:

r prta go

Ready (MacKenzie)

x inc klm ih001

```

```
sc_command r bk start_dump sys_dirs klm 1 60
sc_command r bk ih001
bk r 941 0.324 19.431 152
RCP: Attached tapa_01 for Backup.SysDaemon.z
RCP: Note (tapa_01) -ih001,sys
RCP: Unloading volume from device tapa_01
RCP: Mount Reel ih001 with ring on tapa_01 for Backup.SysDaemon.z
bk
->bk

x vinc klm

vinc
->vinc
vinc Mounting tape iv001 for writing
RCP: Mount Reel iv001 with ring on tapa_01 for Volume_Dumper.Daemon
```





# SECTION 17

## SHUTTING THE SYSTEM DOWN

The system must be shut down when there is a new version of Multics to install, and when the machines need to be powered off for maintenance. In addition, your site may shut the system down on some regular basis: monthly, weekly, or perhaps even daily.

### TO DO A SCHEDULED SHUTDOWN

1. Type:

```
down <shutdown time> <back up time> <message>
```

for example:

```
down 05:45 09:00 Regularly scheduled shutdown
```

where "05:45" is the time of the shutdown, "09:00" is the time when the system will come back up, and "Regularly scheduled shutdown" is a message of your choice. This command may be typed well in advance of when you want the shutdown to occur.

2. The system will respond with:

```
down: Next shutdown from 05:45 to 09:00 Regularly
      scheduled shutdown
down: auto shutdown will be at 06/01/83 1745.0
down: auto abs stop will be at 06/01/83 1725.0
```

3. Twenty minutes before the scheduled time of the shutdown, the system will stop the absentee jobs. You will get a message like this:

```
stop: auto abs stop
```

4. No more than 20 minutes later, you will get a message like this:

```
as abs: All absentee processes have run to completion.
```

5. When the system begins to shut down, you will get messages like the following:

```
as stop: Automatic shutdown
as word: shutdown message: Multics is shutting down.
as bump: MacKenzie.Operator bumped
as bump: User5.ProjectB bumped
as stop: all users have been given 3 minutes to logout
```

6. Users who try to log in will receive the following message:

```
Multics is shutting down
```

Their terminals will be hung up immediately.

Users who are already logged in will receive the following message:

```
*****06/01/83 1545.0 est Wed
From Multics: Multics is shutting down.
You will be logged out in 3 minutes.
*****
```

They will be logged out in 3 minutes. Users with the nobump privilege will be sent a message informing them that an attempt to bump them has been made.

7. To see if users are logging out, type:

```
hmu
```

8. The system will respond with:

```
Multics 11.0: Yoursite, Yourtown, Yourstate
Load = 1.0 out of 110.0 units: users = 1
Absentee users = 2; Maximum absentee users = 10
```

9. After regular users have been automatically logged out by the system, type:

```
who
```

to see if any privileged users are still logged in.

10. The system will respond with:

```
Login      at      TTY   Load   Chan   Group   PNDS   User ID
06/01/83  0957.3  H19   1.0    a.h012  SysProg > DS   B.Multics
```

11. If any are, type:

```
bump <terminal channel>
```

for example:

```
bump a.h012
```

for each one, where "a.h012" is the channel to which the user's terminal is connected.

12. If your site uses the incremental hierarchy dumper, start it for its final pass by typing:

```
r <backup daemon label> wakeup_dump
```

for example:

```
r bk wakeup_dump
```

where "bk" is your site's label for the backup daemon that controls incremental hierarchy dumps.

13. The system will respond with:

```
->bk
```

14. When the incremental hierarchy dump is finished, type:

```
r bk end_dump
```

15. The system will respond with:

```
RCP: Detached tapa_01 from Backup.SysDaemon.z  
RCP: Errors (tapa_01, volume inc-1) = 29  
->bk
```

16. If your site uses the incremental volume dumper, start it for its final pass by typing:

```
r <backup daemon label> wakeup_volume_dump
```

for example:

```
r vinc wakeup_volume_dump
```

where "vinc" is your site's label for the backup daemon that controls incremental volume dumps.

17. The system will respond with:

```
->vinc
```

18. When the incremental volume dump is finished, type:

```
r vinc end_volume_dump
```

19. The system will respond with:

```
RCP: Detached tapa_02 from Volume_Dumper.Daemon.z  
->vinc
```

20. Wait until the printers have finished, or stop each printer driver by typing:

```
r <printer driver label> halt
```

for example:

```
r prta halt
```

Repeat this command for each printer driver that is running.

21. When the current request finishes, the system will respond with:

```
prta prta driver: All devices are halted
prta Enter command:
```

22. If you've already issued an abs stop command (or one has been issued automatically as a result of your issuing the down command), wait until you get the message saying all absentee processes have run to completion. (See Step 4.) If you haven't issued an abs stop command yet (because you're doing a nonscheduled shutdown and there wasn't time) or you can't wait for the message, bump all remaining absentee processes by typing:

```
abs stop now
```

23. The system will respond with:

```
abs: Bumping all remaining absentee processes.
```

then with:

```
as abs: All absentee processes have run to completion.
```

24. Log out all of the daemons by typing:

```
logout * *
```

25. The system will respond with messages like the following:

```
RCP: Force Detached prta from 10.SysDaemon.z
RCP: Force Detached puna from 10.SysDaemon.z
as LOGOUT Data_Management.Daemon dmn dmn1 0:11 $0.97 (logout)
as LOGOUT 10.SysDaemon dmn cord 0:12 $1.07 (logout)
as LOGOUT Backup.SysDaemon dmn bk 0:13 $1.17 (logout)
as LOGOUT 10.SysDaemon dmn prta 0:14 $1.27 (logout)
as LOGOUT Volume_Dumper.Daemon dmn vinc 0:15 $1.37 (logout)
```

26. When all daemons are logged out, type:

```
shut
```

where "shut" is the short form of the shutdown command.

27. The system will respond with:

```
as act_ctl_: shutdown, 57 0.28 0.28 1.57 0.09 0:56.1 $1128.21
shutdown complete
```

then with:

```
bce (boot) 1703.9:
```

This is the BCE ready message, which indicates that the system has returned to BCE level.

### TO DO A NONSCHEDULED SHUTDOWN

Your system administrator or system maintainer may sometimes ask you to do a nonscheduled shutdown. If you have more than 20 minutes to spare, begin with step 1. Otherwise, begin with step 2.

1. Type:

```
abs stop
```

2. Type:

```
stop
```

This is equivalent to typing:

```
word shutdown
bump * * 3
```

3. Follow steps 5-27 described under "To Do a Scheduled Shutdown."

### TO SHUT DOWN IMMEDIATELY

In the case of an emergency, for example a fire, you will need to know how to shut the system down immediately.

1. Type:

```
shut
```

where "shut" is the short form of the shutdown command.

2. The system will respond with:

```
Shutdown: 73 users still on. Do you want to
shut down?
```

3. Type:

```
yes
```

## PROBLEMS

1. If shutdown fails (i.e., you don't receive the "shutdown complete" message), the system may return to BCE on its own and perform recovery procedures automatically. If not, you must force the system to return to BCE, as described in Section 36 under "To Return to BCE." Then you must perform the recovery procedures manually, as described in Section 37 under "To Recover the System."
2. If a volume can't be shut down because of a hardware problem with the drive it's mounted on, the audible alarm will sound and you will receive this message:

```
disk_emergency: dska_04 inoperative; shutdown of
dska_04 suspended
```

The system will continue to try to shutdown all other volumes. When it's finished, you will receive this message:

```
Shutdown complete except for drives suspended
```

You should try to ready the drive or do whatever else is necessary to make it operative. Then perform an emergency shutdown, as described in Section 37 under "Performing ESD."

3. If a regular shutdown gives you a message like this:

```
Shutdown: 3 users still on. Do you want to shut down?
```

answer "no." Ask your system administrator what action you should take to log out the remaining processes. Never answer "yes" to this question, except in the case of an emergency.

4. To reset the time of a scheduled shutdown, just repeat step 1 of "To Do a Scheduled Shutdown." The new shutdown time you type will replace the old one.
5. To cancel a scheduled shutdown, type:

```
down 0
```

## SECTION 18

# POWERING OFF THE MACHINES

The machines aren't usually powered off unless they need maintenance or Multics is going to be shut down for a long period of time.

### TO POWER OFF THE MACHINES

1. If the bootload console is a 6001/6004, press the OFFLINE button, then the POWER OFF button. If it's a 6601, push the breaker on the back of the Rosy printer over, and slide the POWER switch on the back of the screen to OFF.
2. On each PRU1200/1600 printer, press the POWER OFF button on the front. Then on each printer of any kind, push down the breaker on the back of it.
3. On each 451 disk drive, push the STOP button to cycle down the disks. Then push down the breaker, located on the bottom left of the back side of the drive.
4. On each 500 and 501 disk drive, go to the rear of the unit. Set the ONLINE/OFFLINE switch for both side A and side B to OFF, then to ON. Push the STOP button to cycle down the disks. Wait 15 minutes, then press the POWER OFF button.
5. Remove the tapes from the tape drives. Shut the doors. Press the STANDBY button and then press the POWER OFF button on the front of each drive (except 500 and 610 drives). Push down the breaker on the back side of each drive.
6. On the 8200 head-of-string, remove the floppy diskette from the diskette reader. Press the POWER OFF button. (This button is located in the upper right hand corner of the FE panel.)
7. Press the SUBSYSTEM POWER OFF button on the 3380/3381 storage director. Press the POWER OFF button for each controller on the 3380/3381 device controller.
8. Press the POWER OFF button on each unit record MPC.
9. Press the POWER OFF button on each disk MPC.
10. Press the POWER OFF button on each tape MPC.
11. Press the POWER OFF button on each FNP.



12. Press the POWER OFF button on each IOM.
13. Press the POWER OFF button on each SCU.
14. Press the POWER OFF button on each CPU.

PART V  
EVERYDAY OPERATIONS -- STORAGE DEVICES

## SECTION 19

# MANAGING TAPES

In the examples in this section, when you see something like this:

```
tapa_01
```

"tap" specifies what kind of device is involved (tape vs disk), "a" specifies which subsystem is involved (a vs b, etc.), and "01" specifies which drive is involved (01 vs 02, etc.).

### TO MOUNT A TAPE

Tapes are kept in slots in a tape rack. The name of the slot should match the name on the "sticky" tape label and the tape ring label, which should match the name on the magnetic tape label. Foreign tapes (tapes which come from other sites) may have names which are different from the names at your site, and may end up in slots with names which don't match. If you have questions about foreign tapes, ask your system administrator for assistance.

1. When a user wants to use a tape, the first thing he does is ask the system to attach a tape drive for him. When the system has made the drive available to the user's process, you receive several messages from RCP on the bootload console. The first message looks like this:

```
RCP: Attached tapa_05 for User2.ProjectA
```

where "tapa\_05" is the tape drive the system made available to the user's process, and "User2.ProjectA" is the user. This message alerts you to the fact that you may have to mount a tape soon.

2. The second message you receive may make the audible alarm sound. It is usually one of the following:

```
RCP: Using Reel ut565 on tapa_05 for User2.ProjectA.
```

or:

```
RCP: Reready tapa_05.
```

or:

```
RCP: Mount Reel ut565 with ring on tapa_05 for  
User2.ProjectA.
```

where "ut565" is the tape reel name, "with ring" indicates that the reel should be mounted with a write ring, "tapa\_05" is the drive on which the reel should be mounted, and the name of "User2.ProjectA" is the name of the user requesting the mount. Any reel currently ready on the specified drive that isn't the reel requested by the user is automatically demounted.

### Beginning the Procedure

3. If you get the first message, it means that the requested tape is already mounted on the correct drive. Continue with step 20.
4. If you get the second message, it means that the requested tape is already mounted on the correct drive, but the drive has dropped into standby. (The STANDBY light is on.) Ready the drive, and continue with step 20. (See "To Ready a Tape Drive" next in this section.)
5. If you get the third message, continue with step 6.
6. Look for the requested tape reel. If you can't find it or it isn't available, continue with step 18 to deny the tape. If you locate it, check to make sure that the user is allowed to use it. If he isn't, continue with step 18 to deny the tape. If he is, insert or remove a write ring as specified in the mount message. Mount the tape on the specified drive, and ready the drive. (See "To Ready a Tape Drive" next in this section.) If the drive is down, continue with step 18 to deny the tape.
7. The system first checks to see that its instructions about the write ring have been obeyed. If you mounted the tape reel incorrectly, the system unloads the reel and gives you this message:  
  
RCP: Remount Reel ut565 with ring on tapa\_05.
8. Fix the write ring, remount the tape, and ready the drive. (See "To Ready a Tape Drive" next in this section.)
9. The system then checks the magnetic tape label to see that the correct tape has been mounted. If the label is good, the system will go ahead and use the tape. Continue with step 20. If the label is invalid, unrecognizable, or doesn't match the user's request, the system sounds the audible alarm and gives you a message like this:  
  
RCP: Authenticate tapa\_05. It has no label.  
  
or like this:  
  
RCP: Authenticate tapa\_05. It has Multics label ut566.
10. If you get the first message, it means the system can't determine from the magnetic tape label that the correct reel is mounted on the drive. In this case, you should authenticate the tape by continuing with step 12.

11. If you get the second message, it often means that you made a mistake and mounted the wrong tape. In this case, you should mount the correct tape and ready the drive. (See "To Ready a Tape Drive" next in this section.) Then return to step 7.

### Authenticating a Tape

12. Authenticating a tape means verifying that the correct reel has been mounted. The first thing you should do is check the "sticky" tape label. If it doesn't say what you think it should say, return to step 6.
13. If the "sticky" tape label is right, and the tape has an authentication sticker, type:

```
x auth <tape drive> <auth. code>
```

for example:

```
x auth tapa_05 abc
```

where "abc" is the three letter code you see on the sticker.

14. If the tape reel doesn't have an authentication sticker, type:

```
x auth tapa_05 ***
```

Depending on how your system administrator has set up the RCP modes, the system may accept this as validation.

15. The system checks to see that the authentication you supply belongs with the requested tape reel. If it does, continue with step 20. If the authentication is invalid, the system demounts the reel and gives you this message:

```
RCP: Remount Reel ut565 with ring on tapa_05
```

16. Mount the correct tape and ready the drive. (See "To Ready a Tape Drive" next in this section.) Then return to step 7.
17. If the correct tape can't be located or isn't available, if the user has incorrect access to it or you can't authenticate it, or if the specified tape drive is down, you indicate this to the user by denying the tape.

### Denying a Tape

18. Denying a tape is sometimes called *killing a mount request*. Type:

```
x deny tapa_05
```

19. The system will respond with:

RCP: Force unassigned tapa\_05 from User2.ProjectA

The user process which requested the tape mount will receive an error message. If you deny a tape, stop here. Do not do steps 20-23.

### Finishing the Procedure

20. Once the system is sure that the correct reel has been mounted, it makes the tape available to the user and allows the user's process to proceed.
21. When the user's process is finished with the tape and the drive has been released, you get another message on the bootload console. The message looks like this:

RCP: Detached tapa\_05 from User2.ProjectA

22. Depending on how your system administrator has set up the RCP modes, the tape reel may be automatically demounted. Do not demount the tape yourself.
23. If you don't satisfy a mount request within four minutes, the audible alarm will sound and you'll get this message:

RCP: Check Mount of ut565 for writing on tapa\_05 for  
User2.ProjectA

This message will be repeated every four minutes as long as the mount is still pending, or authentication hasn't been done.

### TO READY A TAPE DRIVE

1. On a 500 tape drive, press the LOAD button, wait for the BOT light to go on, and then press the READY button.
2. On a 600, 610 or 630 tape drive, press the START button.
3. On an 8200 tape drive, press the LOAD/REWIND button and then the START button, or just press the START button. Pressing LOAD/REWIND first allows the drive to be loaded, but not to be made ready until you press START.

### TO REWIND A TAPE

1. If the tape is mounted on a 500 tape drive, press the STANDBY button, then the REWIND button, then the READY button on the tape drive.
2. If the tape is mounted on a 600, 610 or 630 tape drive, press the STOP button, then the REWIND button, then the START button on the tape drive.
3. If the tape is mounted on an 8200 tape drive, press the RESET button, then the LOAD/REWIND button, then the START button.

## TO DEMOUNT A TAPE

The system will demount a tape if it needs the drive the tape is on. Normally, it's not a good idea for you to demount a tape unless, for some reason, you need the drive. If the tape is used again, the system will look for it in the same place. If you've demounted it, the system won't be able to find it, and you'll have to go through the whole mount procedure again.

1. Type:

```
unload tape_vol <tape name>
```

for example:

```
unload tape_vol ut565
```

where "ut565" is the name of the tape you want to demount.

2. The system will respond with:

```
RCP: Unloading reel ut565 from device tapa_05
```

where "tapa\_05" is the drive on which the reel is mounted.

3. If the system can't demount the tape for some reason, the audible alarm will sound, and you'll get this message:

```
RCP: Manually unload reel ut565 from device tapa_05
```

In this case, you will have to demount the tape yourself.

4. Return the tape to the library for storage.
5. If you are asked to demount all the tapes before shutting the system down, type:

```
unload tape_vol -all
```

## TO PRELOAD A TAPE

Any time you know in advance that a tape will need to be mounted, it's a good idea to preload it. That way, you can be sure that the tape will be available when it's needed, even if you're away from the machine room then.

If you're going to be running in unattended mode, you should preload tapes so they will be available for backup. (See Section 31, "Setting Attended/Unattended Mode and Manual/Automatic Mode.") When you do this, the tapes are mounted and the tape drives are not released, so when the system is done using the tapes, they aren't demounted.

1. Type:

```
preload <tape drive> <tape name> -user <user name>
```

for example:

```
preload tapa_01 pl123 -user Backup.SysDaemon
```

where "tapa\_01" is the name of the tape drive on which the tape is to be preloaded, "pl123" is the name of the tape, and "Backup.SysDaemon" is the name of the user for whom the preloading is being done.

2. The system will respond with:

```
RCP: Preload reel K1234 on device tapa_01
```

3. Mount the tape you want to preload on the specified drive, and ready the drive.

4. When the system uses the preloaded tape, you'll get this message:

```
RCP: Using Reel K1234 on tapa_01 for Backup.SysDaemon
```

## TO ADD A TAPE DRIVE

Adding a tape drive means adding a drive to the Multics configuration. Usually, the reason you do this is because the drive was broken and CSD has fixed it. You may also do it because the system is in unattended mode and a system administrator or maintainer needs to use a tape. Rather than putting the system in attended mode, which allows all users to use tapes, you can leave it in unattended mode and add a tape drive, which allows the system person alone to use a tape. (For information about attended and unattended mode, see Section 31.)

1. Type:

```
rcf add dv <tape drive>
```

for example:

```
rcf add dv tapa_01
```

where "rcf" is the short form of the reconfigure command, "dv" is short for device, and "tapa\_01" is the drive you want to add.

2. The system should respond with:

```
RCP: Added device tapa_01
```

If it doesn't, there may be something wrong with the drive. When you attempt to add a drive to the configuration, the system tries to read certain information (like speed and density capability) contained in the drive. If this information isn't available, the system will not add the drive. When this happens, you should contact CSD for assistance.



## TO DELETE A TAPE DRIVE

Deleting a tape drive means deleting a drive from the Multics configuration. Usually, the reason you do this is because the drive is broken.

1. Type:

```
rcf dl dv <tape drive>
```

for example:

```
rcf dl dv tapa_01
```

where "rcf" is the short form of the reconfigure command, "dl" is short for delete, "dv" is short for device, and "tapa\_01" is the drive you want to delete.

2. If a user has the drive attached, the system will wait until the user releases it. Then the system will respond with:

```
RCP: Deleted device tapa_01
```

## TO GET INFORMATION ABOUT TAPE MOUNTS

1. To list pending tape reel (and user I/O disk pack) mounts, type:

```
rcp list -mnt -lg
```

where "mnt" is short for mounts, and "lg" is short for long.

2. The system will respond with:

```
1 pending mount  
Mount of ut565 with ring pending on tapa_05 for User2.ProjectA
```

3. To list the status of tape drive 05, type:

```
rcp list -dv tapa_05
```

where "dv" is short for device.

4. The system will respond with:

```
tape_drive device: tapa_05  
State      = assigned  
Time       = 06/01/83 1640.4 est Wed  
User       = User2.ProjectA.a  
Mount pending for volume: ut565
```

## PROBLEMS

1. If a tape drive drops into standby (the STANDBY light goes on) and the tape mounted on it gets slack, the tape has probably lost vacuum. The user may release the drive and try to remount the tape. If he does, the system will find the tape on the drive, but won't be able to read the label. You'll get this message on the bootload console:

```
RCP: Reready tapa_05
```

Ready the drive, put it back into standby by pressing the STANDBY button, and rewind the tape. Notify the tape's user of the problem by typing:

```
w <user name> <"message">
```

for example:

```
w User2.ProjectA "Problems with your tape. Call x7739."
```

where "w" is the short form of the warn command, "User2.ProjectA" is the name of the user, and "Problems with your tape. Call x7739" is a message of your choice.

2. If a tape goes off the end of the reel, rethread it, then rewind it.
3. If a tape wiggles, growls, or otherwise doesn't go smoothly, there's probably something wrong with the drive. Type:

```
rcf dl dv tapa_01
```

When the person using the tape releases the drive, the drive will be deleted. Notify CSD so they can run T & D on the drive.

4. If a drive is in STANDBY when you try to authenticate a tape, you'll have to rewind the tape, ready the drive, then try again to authenticate the tape.

## GOOD IDEAS

1. We recommend that you clean the tapes periodically.
2. We also recommend that you clean the tape drives. During normal usage, this should be done every eight hours.
3. All tapes should be labelled correctly and visibly.

## SECTION 20

# MANAGING USER I/O DISKS

In the examples in this section, when you see something like this:

```
dsk_a_01
```

"dsk" specifies what kind of device is involved (disk vs tape), "a" specifies which subsystem is involved (a vs b, etc.), and "01" specifies which drive is involved (01 vs 02, etc.).

When you see something like this:

```
dsk_a_01b
```

"b" specifies which subvolume is involved (a vs b, etc.). The subvolume letter defines the correct area of the device to be addressed by the system when the system is operating on a physical volume (as opposed to an entire device).

### TO MOUNT A USER I/O DISK PACK

Usually, when you get a request to mount a user I/O disk, the pack involved is demountable (e.g., a 451). The procedure described below assumes that this is the case. However, sometimes the pack involved isn't demountable (e.g., it's a 500 or 501). When this happens, the procedure is a little different. You don't have to locate the pack and mount it on the drive because, of course, it's already mounted. But you still have to set the protect status, and you may still have to authenticate it.

1. When a user wants to use a user I/O disk pack, the first thing he does is ask the system to attach a disk drive for him. When the system has made the drive available to the user's process, you receive several messages from RCP on the bootload console. The first message looks like this:

```
RCP: Attached dsk_a_02 for User6.ProjectB
```

where "dsk\_a\_02" is the disk drive the system made available to the user's process, and "User6.ProjectB" is the user. This message alerts you to the fact that you may have to mount a user I/O disk pack soon.

2. The second message you receive may make the audible alarm sound. It is usually one of the following:

```
RCP: Using Pack ud202 on dsk_a_02 for User6.ProjectB
```

or:

```
RCP: Reready dsk_a_02.
```

or:

RCP: Mount Pack ud202 with protect on dska\_02 for  
User6.ProjectB.

where "ud202" is the disk pack name, "with protect" indicates that the pack should be mounted with write protect, "dska\_02" is the drive on which the pack should be mounted, and "User6.ProjectB" is the name of the user requesting the mount. Any pack currently ready on the specified drive is automatically demounted.

3. If you get the first message, it means that the requested pack is already mounted on the correct drive. Continue with step 17.
4. If you get the second message, it means that the requested pack is already mounted on the correct drive, but the drive has dropped into standby. (The READY light is off.) Ready the drive by pressing the START button, and continue with step 17.
5. If you get the third message, continue with step 6.
6. Locate the requested disk pack. Set the protect status as specified in the mount message by turning the PROTECT switch on the specified drive to ON or OFF. Mount the pack on the specified drive, and ready the drive by pressing the START button.
7. The system first checks to see that its instructions about the protect status have been obeyed. If you mounted the disk pack incorrectly, the system unloads the pack and gives you this message:

RCP: Turn protect switch ON on drive dska\_02 and then push  
Operator Interrupt on MPC.

8. Fix the protect status and push the OPERATOR INTERRUPT button on the disk controller as specified in the message.
9. The system then checks the disk label to see that the correct pack has been mounted. If the label is good, the system will go ahead and use the pack. Continue with step 17.
10. Occasionally, the system won't be able to determine if the label is good, and will ask you to authenticate the mount. It will sound the audible alarm, and give you a message like one of the following:

RCP: Authenticate dska\_02 for User6.ProjectB. It has  
i0 label.

where "IO" is the label type, or:

RCP: Authenticate dska\_02 for User6.ProjectB. It has  
i0 label ud203.

where "IO" is the label type, or:

RCP: Authenticate dska\_02 for Bongo.SysMaint. It has Storage System label sd203.

where "Storage System" is the label type, or:

RCP: Authenticate dska\_02 for Bongo.SysMaint. It has copy of Storage System label sd203.

where "copy of Storage System" is the label type, or:

RCP: Authenticate dska\_02 for Bongo.SysMaint. It has UnReGistered label. (User requested volume ud202.)

where "UnReGistered" is the label type, or:

RCP: Authenticate dska\_02 for User6.ProjectB. It has UnReaDable label. (User requested volume ud202.)

where "UnReaDable" is the label type.

If the disk is an MSU3380 or MSU3381, the system displays a set of messages like

```
RCP: Authenticate dska_00a for WStrunk.  
RCP: It has a copy of Storage System label.  
RCP: root2 on subvol a, pub01 on subvol b, rel03 on subvol c.
```

11. In each case, you must visually check the disk label to see that the correct pack is mounted.
12. If you're able to verify that the correct pack has in fact been mounted, you must authenticate it. If the pack's label type is "IO," type:

```
x auth dska_02 io
```

If the pack's label type is "Storage System," you should not authenticate it unless the user is a system maintainer or has official permission to use it. (What constitutes "official permission" will depend on your site's own policies.) If you're sure it's OK for the user to use the pack, type:

```
x auth dska_02 ss
```

If the pack's label type is "copy of Storage System," treat it like a Storage System pack. Type:

```
x auth dska_02 ss
```

If the pack's label type is "UnReGistered," treat it like a Storage System pack. Type:

```
x auth dska_02 urg
```

If the pack's label type is "UnReaDable," treat it like an IO pack. Type:

```
x auth dska_02 urd
```

13. The system will respond with a message like this:

```
RCP: Mounted IO volume ud202 on dska_02 for user I/O.
```

Continue with step 17.

14. If you discover that the wrong pack has been mounted, mount the correct pack with the specified protect status, and ready the drive by pressing the START button. Then return to step 7.
15. If the user doesn't have permission to use the pack he requested, you must deny the mount. (This is known as *killing a mount request*.) Type:

```
x deny dska_02
```

16. The system will demount the pack and respond with:

```
RCP: Force unassigned dska_02 from User6.ProjectB
```

The user process which requested the disk mount will receive an error message. If you deny a disk, stop here. Do not do steps 17-20.

17. Once the system is sure that the correct pack has been mounted, it makes the disk available to the user and allows the user's process to proceed.
18. When the user's process is finished with the disk, you get another message on the bootload console. The message looks like this:

```
RCP: Detached dska_02 from User6.ProjectB
```

19. The disk drive is automatically demounted.
20. Pending mount messages are repeated every two minutes.

## TO READY A DISK DRIVE

1. Press the START button. The green READY light will go on in about 30 to 60 seconds.

## TO CONVERT A DISK DRIVE FROM USER I/O USE TO STORAGE SYSTEM USE

A drive is only available for one kind of usage at a time. The process of converting a drive from one kind of usage to another is sometimes called *assigning a drive*.

1. Type:

```
sdu <disk drive> ss
```

for example:

```
sdu dska_03 ss
```

where "sdu" is the short form of the set\_drive\_usage command, "dska\_03" is the disk drive you want to convert, and "ss" is short for storage system.

2. The system will respond with:

```
RCP: Consigned dska_03 to storage system.
```

## TO CONVERT A DISK DRIVE FROM STORAGE SYSTEM USE TO USER I/O USE

1. Type:

```
sdu <disk drive> io
```

for example:

```
sdu dska_03 io
```

where "sdu" is the short form of the set\_drive\_usage command, "dska\_03" is the disk drive you want to convert, and "io" is short for user I/O.

2. The system will respond with:

```
RCP: Acquired dska_03 from storage system.
```

## TO ADD A DISK DRIVE

Adding a disk drive means adding a drive to the Multics configuration. The drive is added as a user I/O device. Usually, the reason you do this is because the drive was broken and CSD has fixed it. \*

1. Type:

```
rcf add dv <disk drive>
```

for example:

```
rcf add dv dska_01
```

where "rcf" is the short form of the reconfigure command, "dv" is short for device, and "dska\_01" is the drive you want to add.

2. The system will respond with:

```
RCP: Added device dska_01
```

## TO DELETE A DISK DRIVE

Deleting a disk drive means deleting a drive from the Multics configuration. Usually, the reason you do this is because the drive is broken.

1. Type:

```
rcf dl dv <disk drive>
```

for example:

```
rcf dl dv dska_01
```

where "rcf" is the short form of the reconfigure command, "dl" is short for delete, "dv" is short for device, and "dska\_01" is the drive you want to delete.

2. If a user has the drive attached, the system will wait until the user releases it. Then the system will respond with:

```
RCP: Deleted device dska_01
```

## TO GET INFORMATION ABOUT DISK MOUNTS

1. To list pending user I/O disk pack (and tape reel) mounts, type:

```
rcp list -mnt -lg
```

where "mnt" is short for mounts, and "lg" is short for long.

2. The system will respond with:

```
1 pending mount
Mount of ud202 with protect pending on dska_02 for
User6.ProjectB
```

3. To list the status of disk drive 02, type:

```
rcp list -dv dska_02
```

where "dv" is short for device.



4. The system will respond with:

```
disk_drive device: dska_02
state      = assigned
time      = 06/01/83 1640.0 est Wed
user      = User6.ProjectB.a
Mount pending for volume: ud202
```



## SECTION 21

# MANAGING STORAGE SYSTEM DISKS

In the examples in this section, when you see something like this:

```
dsk a_01
```

"dsk" specifies what kind of device is involved (disk vs tape), "a" specifies which subsystem is involved (a vs b, etc.), and "01" specifies which drive is involved (01 vs 02, etc.).

When you see something like this:

```
dsk a_01b
```

"b" specifies which subvolume is involved (a vs b, etc.). The subvolume letter defines the correct area of the device to be addressed by the system when the system is operating on a physical volume (as opposed to an entire device).

### TO MOUNT A PUBLIC LOGICAL VOLUME

To mount a logical volume, you must mount all of the physical volumes (disk packs) which constitute it.

The system tries to find the packs where they were the last time it used them. If it finds them there, it uses them there. If this is what you want the system to do, follow the procedure described under "Letting the System Decide Where to Mount Packs." The system will tell you where to mount the packs.

If you don't want the system to look for the packs where they were the last time, because you want it to use them somewhere else, follow the procedure described under "Specifying Where Packs Are Mounted." The system will let you specify where the packs are mounted.

### Letting the System Decide Where to Mount Packs

1. Type:

```
alv <logical volume>
```

for example:

```
alv Public
```

where "alv" is the short form of the add\_lv command.

2. The system will respond with a mount request of the form:

```
Mount pack publ on dska_06
```

for each physical volume that must be mounted, where "publ" is the name of the disk pack (physical volume), and "dska\_06" is the name of the disk drive.

3. For 451 disks, mount the requested pack on the requested drive. Ready the drive by pressing the START button.

For 500/501 disks, ready the drive on which the requested pack is located by pressing the START button.

4. Type:

```
av <disk pack> <disk drive>
```

for example:

```
av publ dska_06
```

where "av" is the short form of the add\_vol command. For 500/501 disks, make sure the drive specified in the message in step 2 is the correct one. If it isn't, find out which drive is the correct one, and type its name here instead.

5. Repeat steps 3 and 4 for each physical volume for which you receive a mount request.
6. When all of the necessary physical volumes have been mounted, the system will respond with:

```
lv Public mounted
```

### Specifying Where Packs Are Mounted

1. For 451 disks, mount all of the physical volumes (disk packs) which constitute the logical volume on your chosen disk drives. Ready each drive by pressing its START button.

For 500/501 disks, ready all of the drives on which the physical volumes (disk packs) which constitute the logical volume are located by pressing the START button on each one.

2. For each physical volume that you've mounted, type:

```
av <disk pack> <disk drive>
```

for example:

```
av publ dska_05
```

where "av" is the short form of the add\_vol command, "publ" is the name of the disk pack (physical volume), and "dska\_05" is the name of the disk drive.

This page intentionally left blank.

3. When you've added all of the physical volumes, type:

```
alv <logical volume>
```

for example:

```
alv Public
```

where "alv" is the short form of the add\_lv command.

4. The system will respond with:

```
lv Public mounted
```

## TO MOUNT A PRIVATE LOGICAL VOLUME

1. When a user wants to use a private logical volume, he types the following command in *his* process:

```
alv <logical volume>
```

for example:

```
alv Private
```

where "alv" is the short form of the attach\_lv command (when it's typed in a user's process).

2. The system responds by beginning the process of mounting the logical volume. You get a message on the bootload console, and the audible alarm sounds. The message looks like this:

```
Mount logical volume Private for User5.ProjectB
```

where "Private" is the name of the logical volume, and "User5.ProjectB" is the name of the user.

3. Mounting the logical volume involves mounting each of the physical volumes which constitute it. The system prints a mount request of the form:

```
Mount pack prv1 on dska_07
```

for each physical volume that must be mounted, where "prv1" is the name of the disk pack (physical volume), and "dska\_07" is the name of a suggested disk drive.

4. For 451 disks, mount the requested pack on the suggested drive (or another drive, if you want), and ready it by pressing the START button.

For 500/501 disks, ready the drive on which the requested pack is located by pressing the START button.

5. Type:

```
av <disk pack> <disk drive>
```

for example:

```
av priv1 dska_07
```

where "av" is the short form of the add\_vol command. For 451 disks, if you used a drive other than the suggested one, type its name here instead of the name of the suggested drive.

For 500/501 disks, make sure the drive specified in the message in step 3 is the correct one. If it isn't, find out which drive is the correct one, and type its name here instead.

6. Repeat steps 4 and 5 for each physical volume for which you receive a mount request.
7. When all of the necessary physical volumes have been mounted, the system will respond with:

```
private lv Private mounted
```

8. If you don't satisfy a mount request within four minutes, the audible alarm will sound and you'll get this message:

```
RCP: Check mount of logical volume Private for User5.ProjectB  
priv1 (dska_07), priv2 (dska_04), priv3 (dska_05)
```

where "priv1", "priv2", and "priv3" are disk packs, and "dska\_07", "dska\_04", and "dska\_05" are the drives on which they should be mounted.

This message will be repeated every four minutes as long as the mount is still pending.

#### TO READY A DISK DRIVE

1. Press the START button. The green READY light will go on in about 30 to 60 seconds.

#### TO CANCEL A STORAGE SYSTEM DISK MOUNT

1. If for any reason you need to cancel the pending mount of a logical volume, type:

```
d1v <logical volume>
```

for example:

```
d1v Private
```

where "d1v" is the short form of the del\_lv command.



2. The system will demount each physical volume in the logical volume. You will get a message of the form:

```
demounted privi
```

for each physical volume. Users waiting for the logical volume will receive an error message.

## TO CONVERT A DISK DRIVE FROM STORAGE SYSTEM USE TO USER I/O USE

A drive is only available for one kind of usage at a time. The process of converting a drive from one kind of usage to another is sometimes called *assigning a drive*.

1. Type:

```
sdu <disk drive> io
```

for example:

```
sdu dska_05 io
```

where "sdu" is the short form of the set\_drive\_usage command, "dska\_05" is the disk drive you want to convert, and "io" is short for user I/O.

2. The system will respond with:

```
RCP: Acquired dska_05 from storage system.
```

## TO CONVERT A DISK DRIVE FROM USER I/O USE TO STORAGE SYSTEM USE

1. Type:

```
sdu <disk drive> ss
```

for example:

```
sdu dska_05
```

where "sdu" is the short form of the set\_drive\_usage command, "dska\_05" is the disk drive you want to convert, and "ss" is short for storage system.

2. The system will respond with:

```
RCP: Consigned dska_05 to storage system.
```

## TO ADD A DISK DRIVE AS A STORAGE SYSTEM DRIVE

A disk drive is added as a user I/O device, so you have to convert it to storage system use to add it as a storage system device.

1. Type:

```
rcf add dv <disk drive>
```

for example:

```
rcf add dv dska_01
```

where "rcf" is the short form of the reconfigure command, "dv" is short for device, and "dska\_01" is the drive you want to add.

2. The system will respond with:

```
RCP: Added device dska_01
```

3. Type:

```
sdu <disk drive> ss
```

for example:

```
sdu dska_01 ss
```

where "sdu" is the short form of the set\_drive\_usage command, "dska\_01" is the drive you've added and want to convert, and "ss" is short for storage system.

4. The system will respond with:

```
RCP: Consigned dska_01 to storage system.
```

## TO DELETE A STORAGE SYSTEM DISK DRIVE

To delete a storage system drive, you have to demount the logical volume whose physical volume resides on the drive, demount that physical volume as well as the rest of the physical volumes that constitute the logical volume, and then delete the drive.

1. The logical volume you want to demount may have *process directories* on it. A process directory is a directory which contains segments that are only meaningful during the life of a process. To find out if there are process directories on the logical volume you want to demount, type:

```
list_vols <logical volume> -tt
```

for example:

```
list_vols Public -tt
```

2. The system will respond with:

Records	Left	%	VTOCEs	Left	%	PB/PD	LV Name
387022	52087	13	80640	37060	46	pb pd	Public

3. If the PD flag is on, i.e., you see "pd" under the column headed "PD," there are process directories on the logical volume. To get the process directories off of the logical volume, type:

```
vacate_pdir_volume Public
```

4. The system will respond with:

```
Ready (MacKenzie)
```

5. Now type:

```
dlv Public
```

where "dlv" is the short form of the del\_lv command.

6. Eventually, the system will respond with:

```
demounted lv Public
```

7. The system will demount each of the physical volumes which constitute the logical volume. As it demounts each pack, it will send you this message:

```
demounted publ
```

where "publ" is the name of the disk pack (physical volume) that has been demounted.

8. When the system has demounted all of the physical volumes, type:

```
rcf dl dv <disk drive>
```

for example:

```
rcf dl dv dska_02
```

where "rcf" is the short form of the reconfigure command, "dl" is short for delete, "dv" is short for device, and "dska\_02" is the drive you want to delete.

9. The system will respond with:

```
RCP: Deleted device dska_02
```

## TO GET INFORMATION ABOUT DISK MOUNTS

1. To list pending storage system disk pack mounts, type:

```
ld -mt
```

where "ld" is the short form of the list\_disks command, and "-mt" is short for -mounts.

2. The system will respond with:

```
public lv Temp mount in progress
```

3. To list the status of disk drive 05, type:

```
rcp list -dv dska_05
```

where "-dv" is short for -device.

4. The system will respond with:

```
disk_drive device: dska_05
state      = assigned
time       = 06/01/83 1642.0 est Wed
user       = Sissy.SysAdmin.a
Mount pending for volume scratch
```

## PROBLEMS

1. If an unprivileged user attempts to use a storage system disk as a user I/O disk, you get a message on the bootload console and the audible alarm sounds. The message looks like this:

```
RCP: Rejected mount of Storage System volume f606
for User5.ProjectB
```

PART VI  
EVERYDAY OPERATIONS -- UNIT RECORD DEVICES

## SECTION 22

# MANAGING I/O DAEMONS -- PRINTERS

Daemon printouts of users' files are called *dprints*. The I/O daemon adds pages at the beginning and the end of dprints. These pages contain extra information printed by the I/O daemon to help you distribute the dprints.

The beginning identification page of a dprint is called the *headsheets*. The ending identification page of a dprint is called the *tail/sheet*. At the bottom of the headsheets, several lines of characters are printed. These are called *banner bars*. (Your system administrator may select not to have these sheets and bars printed.)

As each printer request is received from the I/O coordinator, the printer driver prints a short description of the request in the IO Daemon log. The description looks like this:

```
Request 10001 printer q3: >udd>invoices>Station_A.invoices
    from User.Project.a (for "Heading" at "Destination")
    Time estimate for request 10001: 3.4 minutes
```

where "10001" is the *coordinator request number* of the request, ">udd>invoices>Station\_A.invoices" is the pathname of the request, and "User.Project.a" is the name of the user to whom the request belongs. (The time estimate is shown only if it exceeds 1 minute.) The request is usually printed immediately. When it's completed, another message is printed in the IO Daemon log, giving the charge for the request:

```
Charge for request 10001: $9.20 (7546 lines, 423 pages per copy)
```

The driver then asks the I/O coordinator for the next request.

If your system administrator has set a maximum request time limit, and the estimated processing time of the request exceeds this limit, the description looks like this:

```
Request 10002 printer q3: >udd>invoices>Station_B.invoices
    from User.Project.a (for "Heading" at "Destination")
    **Deferring Request 10002. Printing time estimate: 12.3 minutes
```

When a request is deferred, it gets printed later, when more resources are available. See "To Defer a Printer Request" and "To Reprocess a Deferred Request" later in this section.

## TO START UP THE PRINTER

If the printer daemon was logged in, but wasn't started up during system startup, you can start it up manually. (The coordinator must also be logged in.)

1. Type:

```
x <printer driver label>
```

for example:

```
x prt a
```

2. The system will respond with:

```
prt a Enter command: coordinator, driver, or logout:
-> prt a
prt a
prt a 10 Daemon Driver version: 5.7
prt a
prt a Enter command or device/request_type:
as reply go to prt a when it is ok; reply sample_hs to see
new alignment
cord New driver for device prt a, request type printer
(series = 60000)
prt a
prt a Parameters set for the standard printer request type.
prt a
prt a prt a driver ready at 06/01/83 1246.6 est Wed
prt a
prt a Enter command:
```

3. To start processing requests, type:

```
r prt a go
```

4. The system will respond with:

```
Ready (MacKenzie)
```

## TO FIX THE PAPER IN THE MIDDLE OF A REQUEST

Use this procedure when the paper gets messed up in the middle of a request, or when the printer runs out of paper.

## Beginning the Procedure

1. Interrupt the printer by typing:

```
quit <printer driver label>
```

for example:

```
quit prta
```

2. The system will respond with:

```
Sending quit to 004000242236
prta * QUIT * request in progress
prta Enter command(quit):
```

If there is no request in progress, the QUIT message will just say "prta \* QUIT \*".

3. Type:

```
r prta restart
```

4. The system will respond with a form feed on the printer, and the following messages:

```
prta Restarting request 60001 at copy number 4 of 5
prta Driver positioned at page 3 of the file.
prta
prta Enter command(request):
```

## Fixing the Paper

5. Open the printer door.
6. Declutch the tractors. On a PRU0901/1201 or PRU0903/1203 printer, this means moving the orange switch to the left of the paper (marked PUSH TO CLUTCH/TRACTORS/PUSH TO DECLUTCH) to the correct position. On a PRU1200/1600 printer, this means pushing down the white lever on the left side of the bar behind the paper until the paper can turn.
7. Line up a paper fold that points inward (toward the printer) with the red line at the top of the right hand tractor, by hand-rotating the bar behind the paper.
8. Hit the SKIP button.
9. Reclutch the tractors. On a PRU0901/1201 or PRU0903/1203 printer, this means moving the orange switch to the correct position. On a PRU1200/1600 printer, this means raising the white lever as far as it will go.
10. Hit the SKIP button again, then the START button.



## Finishing the Procedure

11. To check the paper alignment, type:

```
r prta sample_hs
```

12. The system will respond with:

```
prta Enter command(quit):
```

- \* 13. If the paper alignment isn't OK, repeat steps 5 through 12 until it is.

14. If you want the printer to restart where it says it's going to (in this case, page 3), type:

```
r prta p`rint
```

15. If you want it to restart at an earlier page (say, page 2), type:

```
r prta print 2
```

or:

```
r prta print -<number of pages before current page>
```

for example:

```
r prta print -1
```

16. If you want it to restart at a later page (say, page 4), type:

```
r prta print 4
```

or:

```
r prta print + <number of pages after current page>
```

for example:

```
r prta print +1
```

17. If you want it to restart at a different copy, type:

```
r prta copy 3  
r prta print
```

18. After the request has printed, the system will respond with:

```
prta Charge for request 60001: $0.29  
(49 lines, 10 pages per copy)
```

## TO REPRINT A SEGMENT

1. Type:

```
x reprint <segment name> <user name>
```

for example:

```
x reprint >udd>ProjectA>User3>new_pgm User3.ProjectA
```

where ">udd>ProjectA>User3>new\_pgm" is the name of the segment you want to reprint, and "User3.ProjectA" is the name of the user who requested the original dprint.

2. The system will respond with:

```
dprint -q 1 -he REPRINT -ds "User3.ProjectA"  
>udd>ProjectA>User3>new_pgm  
1 request signalled, 0 already in printer queue 1  
prta Request 60002 printer q 1: >udd>ProjectA>User3>new_pgm  
prta From Initializer.SysDaemon.z (for "REPRINT" at  
"User3.ProjectA")  
prta Charge for request 60002: $0.18 (49 lines, 1 pages)
```

3. The user "User3.ProjectA" will get a dprint of the segment ">udd>ProjectA>User3>new\_pgm" with the header "REPRINT". The cost of the dprint will be charged to the system.

## TO CHANGE THE CURRENT REQUEST TYPE PROCESSING

1. Type:

```
r <printer driver label> halt
```

for example:

```
r prta halt
```

2. The printer will stop taking requests after it's done with the current one. The system will respond with:

```
prta prta driver: All devices are halted.  
prta Enter command:
```

3. Type:

```
r prta new_device
```

4. The system will respond with:

```
RCP: Detached prta from IO.SysDaemon.z  
prta Enter command or device/request_type:  
cord Driver logout for device prta  
-> prta
```

5. Type:

```
r <printer driver label> <device name>
  <new request type>
```

for example:

```
r prta prta unlined
```

6. The system will respond with:

```
cord New driver for device prta, request type unlined
  (series = 70000)
RCP: Attached prta for 10.SysDaemon.z
prta
prta Parameters set for the standard printer request type.
prta
prta prta driver ready at 06/01/83 1256.2 est Wed
prta
prta Enter command:
```

7. If you have to change the paper (for example, from lined to unlined), do that now.

8. If you want to check on the paper alignment, type:

```
r prta sample_hs
```

9. The system will respond with:

```
prta Enter command:
```

10. To start processing requests again, type:

```
r prta go
```

11. The system will respond with:

```
Ready (MacKenzie)
```

## TO FIND OUT THE STATUS OF THE PRINTER

1. Type:

```
r <printer driver label> status
```

for example:

```
r prta status
```

2. When the daemon finishes printing the current request, the system will respond with:

```
prta
prta IO Daemon Version: 5.7
prta Device: prta iom ch: prta
prta Request type: printer
prta Status: ready
prta
prta Enter command:
```

3. To start processing requests again, type:

```
r prta go
```

4. The system will respond with:

```
Ready (MacKenzie)
```

#### TO GET INFORMATION ABOUT DEVICES AND REQUEST TYPES (AND DEVICE CLASSES)

1. If you aren't sure what devices and request types (and device classes, if your site uses them) are available, you can list them by typing:

```
r <l/O daemon coordinator label> print_devices
```

for example:

```
r cord print_devices
```

2. The system will respond with something like the following:

Device	Request type	Access name	
* prta	printer	10.SysDaemon	system_high
* prtb			
* prtc			
* prtd			
prta	test	10.SysDaemon	system_high
prtb			
prtc			
prtd			
* puna	punch	10.SysDaemon	system_high
* punc			
prta	unlined	10.SysDaemon	system_high
prtb			
prtc			
prtd			
* rdra	reader_dummy	10.SysDaemon	
* rdrb			
* rdrc			
* rdrd			

The devices are grouped according to the request types they service. An asterisk (\*) before a device indicates that the associated request type is the default for the device. The driver access name and authorization (if any) are given for each request type.

3. To find out what devices and request types (and device classes, if your site uses them) are already running, type:

```
r <I/O daemon coordinator label> list
```

for example:

```
r cord list
```

4. The system will respond with something like the following:

```
device prta is active, request type printer, request 40017
device prtb is active, request type unlined.confidential,
request 20007
device rem1 is active, request type rem1_print_unlined,
request 50002
device rem2.elite is active, request type printer,
request 70003
device rem2.pica is active, request type rem2_pica_unlined,
request 30015
```

## TO GET INFORMATION ABOUT PRINTER REQUESTS

1. To list all of the pending printer requests, type:

```
r prta x ldr
```

where "ldr" is the short form of the `list_daemon_requests` command.

2. The system will respond with a message like the following:

```
printer queue 3: 2 total requests.
```

User	ID	Entry name
User5.ProjectA.a	1) 182329.7	new_version.compout (running)
User2.ProjectB.m	2) 224326.9	output_file

where "182329.7" and "224326.9" are the *match ids* of the requests.

It might also respond with:

```
There are no requests in any printer queue.
```

3. To start processing requests again, type:

```
r prta go
```

4. The system will respond with:

```
Ready (MacKenzie)
```

## TO KILL A PRINTER REQUEST

Killing the current request stops the printer from printing it and places it in the coordinator's saved list. Usually, you should only kill a request if a user calls and asks you to do so. However, you may also have to kill a request if something is obviously very wrong with it (for example, it seems to contain nothing but form feed characters). Note: Killed requests are lost if the coordinator reinitializes.

1. Interrupt the printer by typing:

```
quit <printer driver label>
```

for example:

```
quit prta
```

2. The system will respond with:

```
Sending quit to 004000242236  
prta * QUIT * request in progress  
prta Enter command(quit):
```

3. Type:
 

```
r prta kill
```
4. The system will respond with:
 

```
prta **Request 80001: Operator aborted listing during
processing of request.
prta Processing of request 80001 terminated.
```
5. The driver will continue with the next request.

### TO REPROCESS A KILLED REQUEST

1. Type:
 

```
r <printer driver label> restart <coordinator request number>
```

for example:

```
r prta restart 80001
```

where "80001" is the coordinator request number of the killed request you want to reprocess.

2. The system will respond with:
 

```
cord Restart command received from device prta
cord Restart of request 80001 initiated for request
type printer
cord Device prta switched to series 17000
prta Enter command:
```

3. Type:
 

```
r prta go
```

4. The system will respond with:
 

```
Ready (Mackenzie)
```

### TO DEFER A PRINTER REQUEST

Deferring the current request sends it back to its queue marked "deferred". Usually, things are set up so that a request is automatically deferred if its estimated processing time exceeds a defined limit. But if this isn't the case at your site, or you get another request that needs rush printing, or you just have questions about a request, you may defer it yourself.

1. Interrupt the printer by typing:

```
quit <printer driver label>
```

for example:

```
quit prta
```

2. The system will respond with:

```
Sending quit to 004000242236
prta * QUIT * request in progress
prta Enter command(quit):
```

3. Type:

```
r prta defer
```

4. The system will respond with:

```
prta **Request 80003: Operator deferred request until a
later time.
prta Processing of request 80003 terminated.
```

5. The driver will continue with the next request.

## TO REPROCESS A DEFERRED REQUEST

Deferred requests are reprocessed automatically when the coordinator is next initialized or when you restart the queue. (To restart a queue, type: `r prta restart_q.`) You can also reprocess an individual deferred request, by specifying that it run next.

1. Type:

```
r <printer driver label> next -user <user name> -id
<match id>
```

for example:

```
r prta next -user User6.ProjectB -id 182329.7
```

where "User6.ProjectB" is the name of the user who submitted the request, and "182329.7" is the match id of the deferred request you want to reprocess. (You can find out a request's match id by using the "x ldr" command, as described under "To Get Information About Printer Requests" earlier in this section.)

2. The system will respond with:

```
prta Request found.
prta Enter command:
```

3. Type:

```
r prta go
```



4. The system will respond with:

Ready (MacKenzie)

## TO REINITIALIZE A PRINTER

Printers don't need to be reinitialized very often. About the only time they do is when something happens to the coordinator. In that case, all of the drivers have to be reinitialized.

1. Type:

```
r <printer driver label> reinit
```

for example:

```
r prta reinit
```

2. The system will respond with:

```
RCP: Detached prta from 10.SysDaemon.z
RCP: Attached prta for 10.SysDaemon.z
prta iodd_signal_handler_: Driver starting reinitialization
cord Driver logout for device prta
cord New driver for device prta, request type printer
(series = 90000)
prta
prta Parameters set for the standard printer request type.
prta
prta prta driver ready at 06/01/83 1310.3 est Wed
prta
prta Enter command:
```

3. To start processing requests, type:

```
r prta go
```

4. The system will respond with:

Ready (MacKenzie)

If your site has more than one printer driver, you will need to repeat steps 1 through 4 for each one.

## TO LOG OUT THE PRINTER I/O DAEMON DRIVER

1. If the printer I/O daemon driver seems to be in trouble, you can log it out.  
Type:

```
logout <daemon User_id> <printer driver label>
```

for example:

```
logout 10.SysDaemon prta
```

2. The system will respond with:

```
RCP: Detached prta from 10.SysDaemon.z  
cord Driver logout for device prta  
as LOGOUT 10.SysDaemon dmn prta 0:14 $1.27 (logout)
```

## PROBLEMS

1. If you happen to quit the wrong printer, just type "r prta start" (where "prta" is your site's label for the driver controlling the printer you quit), and it will pick up right where it left off.
2. If there's some reason why the driver is not allowed to operate the device(s) or the request type(s) (or the device class(es), if your site uses them) you specify, the system will respond with one or more error messages explaining the problem, then ask you again for a device name and optional request type.

## GOOD IDEAS

1. Check the printers regularly to see that: the banners are being printed correctly and the print quality is high; the ribbon is good and all positions are printing clearly.



## SECTION 23

# MANAGING I/O DAEMONS -- CARD PUNCHES AND READERS

### TO START UP THE CARD PUNCH

If the punch daemon was logged in, but wasn't started up during system startup, you can start it up manually. (The coordinator must also be logged in.)

1. Type:

```
x <punch driver label>
```

for example:

```
x puna
```

2. The system will respond with:

```
puna Enter command: coordinator, driver or logout:
->puna
puna
puna IO Daemon Driver Version: 5.7
puna
puna Enter command or device/request_type:
cord New driver for device puna, request type punch
      (series = 100000)
puna
puna puna driver ready at 06/01/83 1316.6 est Wed
puna
puna Enter command:
```

3. To start processing requests, type:

```
r puna go
```

4. The system will respond with:

```
Ready (MacKenzie)
```

## TO STOP THE CARD PUNCH

1. Type:

```
x punch_end
```

which will halt the punch driver and detach the card punch device.

2. The system will respond with:

```
puna puna driver: All devices are halted.  
puna Enter command:  
RCP: Detached puna from 10.SysDaemon.z  
puna Enter command or device/request_type:  
->puna  
cord Driver logout for device puna
```

## TO RESTART THE CARD PUNCH

1. Type:

```
x punch_restart
```

which will attach the card punch device and start the punch driver.

2. The system will respond with:

```
cord New driver for device puna, request type punch  
      (series = 110000)  
puna  
RCP: Attached puna for 10.SysDaemon.z  
puna puna driver ready at 06/01/83 1321.6 est Wed  
puna
```

## TO GET INFORMATION ABOUT PUNCH REQUESTS

1. To list all of the pending punch requests, type:

```
r puna x ldr
```

where "ldr" is the short form of the list\_daemon\_requests command.

2. The system will respond with a message like the following:

```
punch queue 1: 1 total request. *
```

User	ID	Entry name
User4.ProjectB.a	1) 255143.9	so_long.punch

It might also respond with:

```
There are no requests in any punch queue.
```

3. To start processing requests again, type:

```
r puna go
```

4. The system will respond with:

```
Ready (MacKenzie)
```

## TO READ CARDS

To read cards, you must make sure the card deck(s) you're going to read are complete, prepare them for reading, and then start up the reader I/O daemon driver.

### Checking Card Decks for Completeness

1. When users submit card decks to operations, the first thing you should do is check to see that the decks are complete and that they follow the correct format.

Card decks may be used for either *bulk data input* or *remote job entry*. A bulk data input card deck contains a program or a data file. A remote job entry card deck contains a series of Multics commands to be run as an absentee job.

The correct format for a bulk data input card deck which contains only the required cards is as follows:

```
++DATA <deck_name> <Person_id> <Project_id>
++PASSWORD <xxxxxxxx>
++INPUT
.
.
(user data cards)
.
.
```

A bulk data input card deck may also contain the following optional cards:

```
++AIM <authorization>
++FORMAT <punch_format> <format_control_modes>
++CONTROL OVERWRITE
```

The correct format for a remote job entry card deck which contains only the required cards is as follows:

```
++RJE <deck_name> <Person_id> <Project_id>
++PASSWORD <xxxxxxx>
++INPUT
.
.
(user absentee commands)
.
.
```

A remote job entry card deck may also contain the following optional cards:

```
++AIM <authorization>
++FORMAT <punch_format> <format_control_modes>
++RJECONTROL <ear_control_args>
++RJEARGS <ear_args>
++EPILOGUE <command_line>
++ABSIN <pathname>
```

- 
2. If the card decks are not in the correct format, return them to the users. If they are, continue with step 3.

### Preparing Card Decks for Reading

- 
- 
3. To prepare users' card decks for reading, you must place *control cards* in front and in back of them. The control cards which go in front of a deck are as follows:

```
++EOF
++UID <uid_string>
```

The ++EOF card is the end-of-file marker. It must be the first card you place in the card reader before you type the read\_cards command (see step 16).

The ++UID card is the unique-id card. It must follow the ++EOF card. You can use any string of 1 to 12 characters (except spaces) for "uid\_string."

The control cards which go in back of a deck are as follows:

```
++EOF
++UID <uid_string>
++END
```

The ++EOF card is the same as the one that goes in front of a deck.

The ++UID card is also the same as the one that goes in front of a deck. Its "uid\_string" must be the same as the "uid\_string" on the ++UID card in front of the deck. When you're preparing more than one deck for reading, the ++UID cards are the way you separate the decks from each other. So, it's a good idea to keep about ten pairs of matching ++UID cards near the card reader, so you can stack several decks in the hopper at one time.

The ++END card is the terminate-reading card. It tells the driver that there are no more decks to be read. If you are preparing more than one deck for reading, you shouldn't put a ++END card in back of every deck. Instead, you should just put one ++END card in back of the last deck.

4. When you create control cards for a deck, make sure you start punching in column one. Also, you should realize that different card readers have different punch card decoding conventions, so the same character may require different punch codes on different readers. For example, a + character may be represented by a 12-8-2 punch on one reader, and a 12-0 on another. Obviously, cards prepared for the first reader do not transmit the same data on the second reader, and may in fact be unreadable. Your system administrator can tell you which decoding conventions work on which readers.

The following is an example of a group of card decks, all ready for reading:

```
++EOF
++UID ZZZZZ
.
(First User Card Deck)
.
++EOF
++UID ZZZZZ
++EOF
++UID ABCDEF
.
(Next User Card Deck)
.
++EOF
++UID ABCDEF
.
.
++EOF
++UID 1234567890
.
(Last user Card Deck)
.
++EOF
++UID 1234567890
++END
```



## Starting Up the Daemon

5. If you're using a Combined Card Unit (CCU), continue with step 6. Otherwise, continue with step 8.

6. Logout the punch I/O daemon driver by typing:

```
logout <daemon User_id> <punch driver label>
```

for example:

```
logout 10.SysDaemon puna
```

7. The system will respond with:

```
RCP: Detached puna from 10.SysDaemon.z  
cord Driver logout for device puna  
as LOGOUT 10.SysDaemon dmn puna 2:11 $4.69 (logout)
```

8. If your site uses AIM and there are ++AIM cards in any of the card decks, sort the decks by authorization.

9. Login the reader I/O daemon driver by typing:

```
login <daemon User_id> <reader driver label>
```

for example:

```
login 10.SysDaemon rdra
```

If your site uses AIM and there are ++AIM cards in any of the card decks, you should specify the authorization of the reader driver by typing this instead:

```
login 10.SysDaemon rdra -auth <authorization>
```

The authorization of the reader driver *must* be the same as the authorization of the first group of decks you want to read. (Note: if your site doesn't use IO.SysDaemon, the daemon User\_id may have the right authorization associated with it, making this unnecessary.)

10. The system will respond with:

```
as LOGIN 10.SysDaemon dmn rdra (create)  
rdra Enter command: coordinator or driver  
-> rdra
```

11. Type:

```
r rdra driver
```

12. The system will respond with:

```
rdra Enter device name and optional request type:  
-> rdra
```

13. Type:

```
r <reader driver label> <device name>
```

for example:

```
r rdra rdra
```

(At most sites, the reader driver label and the device name are the same.)

14. The system will respond with:

```
RCP: Attached rdra for 10.SysDaemon.z  
rdra rdra driver ready at 06/01/83 0855.1 est Wed  
-> rdra
```

15. Place the decks in the hopper.

16. Type:

```
r rdra read_cards
```

17. The system will respond with:

```
rdra Card Input started  
-> rdra
```

18. When the system has read all of the decks, it will respond with:

```
rdra Card Input completed  
rdra Enter command:  
-> rdra
```

19. At this point, if you have more card decks to read and your site doesn't use AIM, or if your site does use AIM, but there are no ++AIM cards in any of the decks, repeat steps 15-18. If you have more card decks to read, your site does use AIM, and there are ++AIM cards in any of the decks, continue with step 20.

20. Logout the reader I/O daemon driver by typing:

```
logout <daemon User_id> <reader driver label>
```

for example:

```
logout 10.SysDaemon rdra
```

21. The system will respond with:

```
RCP: Detached rdra from 10.SysDaemon.z  
cord Driver logout for device rdra  
as LOGOUT 10.SysDaemon dmn rdra 0:12 $1.19 (logout)
```

22. Now repeat steps 9-18. In step 9, login the reader I/O daemon driver with the same authorization as the next group of decks you want to read.

## TO STOP THE CARD READER

1. If the card reader gets into trouble, it may stop itself and print this message:

```
rdra Enter command:
```

If it doesn't, continue with step 2. If it does, continue with step 4.

2. Interrupt the reader by typing:

```
quit <reader driver label>
```

for example:

```
quit rdra
```

3. The system will respond with:

```
sending quit to 005000252236  
rdra * QUIT * request in progress  
rdra Enter command (quit):
```

4. Type:

```
r rdra release
```

5. The system will respond with:

```
rdra Enter command:
```

6. To start reading cards again, type:

```
r rdra read_cards
```

## TO LOGOUT THE PUNCH OR READER I/O DAEMON DRIVER

1. If either the punch or the reader I/O daemon driver seems to be in trouble, you can log it out. Type:

```
logout <daemon User_id> <punch driver label>
```

for example:

```
logout 10.SysDaemon puna
```

or:

```
logout <daemon User_id> <reader driver label>
```

for example:

```
logout 10.SysDaemon rdra
```

2. The system will respond with:

```
RCP: Detached puna from 10.SysDaemon.z  
cord Driver logout for device puna  
as LOGOUT 10.SysDaemon dmn puna 2:11 $4.69 (logout)
```

or:

```
RCP: Detached rdra from 10.SysDaemon.z  
cord Driver logout for device rdra  
as LOGOUT 10.SysDaemon dmn rdra 0:12 $1.19 (logout)
```



## SECTION 24

# MANAGING REMOTE DEVICES AT THE CENTRAL SITE

A *remote device* is a unit record device which is connected to the FNP and communicates with the IOM via the FNP. There are two kinds of remote devices. The first kind isn't actually physically remote from the Multics system, but is considered remote because it's connected to the FNP instead of an MPC. The second kind is physically remote from the Multics system and is often called a *remote station*. It's connected to a *workstation* (another computer which acts as a satellite for Multics) instead of an MPC. The workstation in turn is connected to the Multics FNP.

Remote stations are sometimes called *remote job entry (RJE) stations*. RJE refers to the process whereby standard Multics commands are punched on cards, then copied into an absentee segment. The segment is then submitted as an absentee job. RJE can be done locally or at a remote site. When it's done at a remote site, users submit their card decks to their computer system for execution as jobs on a Multics system. They receive the output of their jobs back on their own system, where it is printed or punched. Technically, a workstation is only an RJE station when it includes a card reader for submitting RJE decks. However, a remote station is often called an RJE station when it only includes a card reader for data decks, or even when it doesn't include a card reader at all.

The only way to run a remote device is by using a *remote driver process*. A remote driver uses *logical devices* to control the physical devices. A logical device is often called a *minor device* and a physical device is often called a *major device*. When a remote device is connected directly to the FNP, it's usually a printer which can print in different formats. Each format has its own queue. One remote driver runs several logical devices, one for each print format. When remote devices are connected to a workstation, they can be printers, card punches or card readers. One remote driver runs several logical devices, one for each remote device.

If your site uses AIM, you may have to specify one or more device classes when you start up a driver.

## TO LOG IN AND START A REMOTE PRINTER/PUNCH

1. Type:

```
login <daemon User_id> <remote printer/punch driver label>
```

for example:

```
login IO.SysDaemon reml
```

If your site uses AIM, you should specify the authorization of the remote driver by typing this instead:

```
login IO.SysDaemon reml -auth <authorization>
```

Your system administrator will tell you what the authorization should be. (Note: if your site doesn't use IO.SysDaemon, the daemon User\_id may have the right authorization associated with it, making this unnecessary.)

2. The system will respond with:

```
as LOGIN IO.SysDaemon dmn reml (create)
reml Enter command: coordinator or driver
-> reml
```

3. Type:

```
r reml driver
```

4. The system will respond with:

```
reml Enter command or device/request type:
-> reml
```

5. If you want the printer/punch to run with the default request type (and the default device class) for each minor device, type:

```
r <remote printer/punch driver label> <major device name>
default
```

for example:

```
r reml reml default
```

(At most sites, the remote printer/punch driver label and the major device name are the same.) Then continue with step 11.

6. If you want the printer/punch to run with a request type (or a device class) other than the default for one or both of the minor devices, type:

```
r <remote printer/punch driver label> <major device name>
```

for example:

```
r reml reml
```

7. The system will respond with:

```
reml Enter request type (or default) for minor
device "printer":
```

8. Type:

```
r <remote printer/punch driver label> <minor device
request type>
```

for example:

```
r reml reml_print_unlined
```

or, if your site uses device classes:

```
r <remote printer/punch driver label>
<minor device request type>.<device class>
```

for example:

```
r reml reml_print_unlined.confidential
```

where "reml\_print\_unlined" is the request type for the logical (minor) device mentioned in the message in step 7.

9. The system will respond with:

```
reml Enter request type (or default) for minor device "punch":
```

10. Type:

```
r <remote printer/punch driver label>
<minor device request type>
```

for example:

```
r reml default
```

or, if your site uses device classes:

```
r <remote printer/punch driver label>
<minor device request type>.<device class>
```

for example:

```
r reml default.public
```

where "default" is the request type for the logical (minor) device mentioned in the message in step 9.



11. The system will respond with:

```
rem1 rem1 driver ready at 06/01/83 1165.2 est Wed
rem1
rem1 Enter command:
```

12. To start processing requests, type:

```
r rem1 go
```

13. The system will respond with:

```
Ready (MacKenzie)
```

### TO LOG IN AND START A REMOTE PRINTER THAT PRINTS IN DIFFERENT FORMATS

In this example, the remote printer prints in two different formats, elite and pica.

1. Type:

```
login <daemon User_id> <remote printer driver label>
```

for example:

```
login IO.SysDaemon rem2
```

If your site uses AIM, you should specify the authorization of the remote driver by typing this instead:

```
login IO.SysDaemon rem2 -auth <authorization>
```

Your system administrator will tell you what the authorization should be. (Note: if your site doesn't use IO.SysDaemon, the daemon User\_id may have the right authorization associated with it, making this unnecessary.)

2. The system will respond with:

```
as LOGIN IO.SysDaemon dmn rem2 (create)
rem2 Enter command: coordinator or driver
->rem2
```

3. Type:

```
r rem2 driver
```

4. The system will respond with:

```
rem2 Enter command or device/request type:
->rem2
```

5. If you want the printer to run with the default request type (and the default device class) for each minor device, type:

```
r <remote printer driver label> <major device name> default
```

for example:

```
r rem2 rem2 default
```

(At most sites, the remote printer driver label and the major device name are the same.) Then continue with step 11.

6. If you want the printer to run with a request type (or a device class) other than the default for one or both of the minor devices, type:

```
r <remote printer driver label> <major device name>
```

for example:

```
r rem2 rem2
```

7. The system will respond with:

```
rem2 Enter request type (or default) for minor device "elite":
```

8. Type:

```
r <remote printer driver label> <minor device request type>
```

for example:

```
r rem2 default
```

or, if your site uses device classes:

```
r <remote printer driver label>  
  <minor device request type>.<device class>
```

for example:

```
r rem2 default.public
```

where "default" is the request type for the logical (minor) device mentioned in the message in step 7.

9. The system will respond with:

```
rem2 Enter request type (or default) for minor device "pica":
```

10. Type:

```
r <remote printer driver label> <minor device request type>
```

for example:

```
r rem2 rem2_pica_unlined
```

or, if your site uses device classes:

```
r <remote printer driver label>  
  <minor device request type>.<device class>
```

for example:

```
r rem2 rem2_pica_unlined.confidential
```

where "rem2\_pica\_unlined" is the request type for the logical (minor) device mentioned in the message in step 9.

11. The system will respond with:

```
rem2 rem2 driver ready at 06/01/83 1165.2 est Wed  
rem2  
rem2 Enter command:
```

12. Ready the minor device you want to be active by typing:

```
r <remote printer driver label> ready <minor device name>
```

for example:

```
r rem2 ready elite
```

13. The system will respond with:

```
rem2 Enter command:
```

14. To start processing requests, type:

```
r rem2 go
```

15. The system will respond with:

```
Ready (MacKenzie)
```

## TO CHANGE FROM ONE PRINT FORMAT TO ANOTHER

In this example, the remote printer prints in two different formats, elite and pica. The example shows how to change from the elite format to the pica format.

1. Type:

```
r <remote printer driver label> halt <minor device name>
```

for example:

```
r rem2 halt elite
```

2. The system will respond with:

```
rem2 rem2 driver: All devices are halted.  
rem2 Enter command:
```

3. Change the type wheels.

4. Type:

```
r <remote printer driver label> ready <minor device name>
```

for example:

```
r rem2 ready pica
```

5. The system will respond with:

```
rem2 Enter command:
```

6. To start processing requests again, type:

```
r rem2 go
```

7. The system will respond with:

```
Ready (MacKenzie)
```

## TO START UP A TYPE I REMOTE STATION

Your system administrator can tell you whether the remote station you're starting up is a Type I station or a Type II station.

1. Type:

```
login <daemon User_id> <remote driver label>
```

for example:

```
login l0.SysDaemon rem3
```

If your site uses AIM, you should specify the authorization of the remote driver by typing this instead:

```
login IO.SysDaemon rem3 -auth <authorization>
```

Your system administrator will tell you what the authorization should be. (Note: if your site doesn't use IO.SysDaemon, the daemon User\_id may have the right authorization associated with it, making this unnecessary.)

2. The system will respond with:

```
as LOGIN IO.SysDaemon dmn rem3 (create)
rem3 Enter command: coordinator or driver
->rem3
```

3. Type:

```
r rem3 driver
```

4. The system will respond with:

```
rem3 Enter command or device/request type:
->rem3
```

5. Type:

```
listen <communications line>
```

for example:

```
listen 16
```

where "16" is the name of a communications line, which in turn specifies a communications channel, over which the remote device communicates with Multics.

6. The system will respond with:

```
Attaching line "16" on channel (a.h001)
```

This indicates that the system is preparing the remote driver to wait for the remote device to log in.

7. When the remote device logs in on the specified line, the system will respond with:

```
Requesting station identifier on line "16"
```

then with:

```
rem3 rem3 driver ready at 06/01/83 1165.2 est Wed
rem3
rem3 Enter command:
```

8. The remote operator will probably start processing requests for you. To start processing requests yourself, type:

```
r rem3 go
```

9. The system will respond with:

```
Ready (MacKenzie)
```

## TO START UP A TYPE II REMOTE STATION

Your system administrator can tell you whether the remote station you're starting up is a Type I station or a Type II station.

1. Type:

```
login <daemon User_id> <remote driver label>
```

for example:

```
login IO.SysDaemon rem4
```

If your site uses AIM, you should specify the authorization of the remote driver by typing this instead:

```
login IO.SysDaemon rem4 -auth <authorization>
```

Your system administrator will tell you what the authorization should be. (Note: if your site doesn't use IO.SysDaemon, the daemon User\_id may have the right authorization associated with it, making this unnecessary.)

2. The system will respond with:

```
as LOGIN IO.SysDaemon dmn rem4 (create)
rem4 Enter Command: coordinator or driver
-> rem4
```

3. Type:

```
r rem4 driver
```

4. The system will respond with:

```
rem4 Enter command or device/request type:
-> rem4
```

5. If you want the remote device to run with the default request type (and device class) for each of its minor devices, type:

```
r <remote driver label> <major device name> default
```

for example:

```
r rem4 rem4 default
```

(At most sites, the remote driver label and the major device name are the same.) Then continue with step 11.

6. If you want the remote device to run with a request type (or a device class) other than the default for any of its minor devices, type:

```
r <remote driver label> <major device name>
```

for example:

```
r rem4 rem4
```

7. The system will respond with:

```
rem4 Enter request type (or default) for minor
device "device1":
```

8. Type:

```
r <remote driver label> <minor device request type>
```

for example:

```
r rem4 request_type_1
```

or, if your site uses device classes:

```
r <remote driver label>
  <minor device request type>.<device class>
```

for example:

```
r rem4 request_type_1.confidential
```

9. The system will respond with:

```
rem4 Enter request type (or default) for minor
device "device2":
```

10. Type:

```
r <remote driver label> <minor device request type>
```

for example:

```
r rem4 default
```

or, if your site uses device classes:

```
r <remote driver label>  
  <minor device request type>.<device class>
```

for example:

```
r rem4 default.public
```

11. It may take a while for the remote station to connect. When it does, the system will respond with:

```
rem4 rem4 driver ready at 06/01/83 1165.2 est Wed  
rem4  
rem4 Enter command:
```

12. If necessary, ready the minor device you want to be active by typing:

```
r <remote driver label> ready < minor device name>
```

for example:

```
r rem4 ready device1
```

13. The system will respond with:

```
rem4 Enter command:
```

14. The remote operator will probably start processing requests for you. To start processing requests yourself, type:

```
r rem4 go
```

15. The system will respond with:

```
Ready (MacKenzie)
```



## TO GET INFORMATION ABOUT DEVICES AND REQUEST TYPES (AND DEVICE CLASSES)

1. If you aren't sure what devices and request types (and device classes, if your site uses them) are available, you can list them by typing:

```
r <l/0 daemon coordinator label> print_devices
```

for example:

```
r cord print_devices
```

2. The system will respond with something like the following:

Device	Request type	Access name	
* prta	printer	10.SysDaemon	system_high
* prtb			
* prtc			
* prtd			
prta	test	10.SysDaemon	system_high
prtb			
prtc			
prtd			
* puna	punch	10.SysDaemon	system_high
* punc			
prta	unlined	10.SysDaemon	system_high
prtb			
prtc			
prtd			
* rdra	reader_dummy	10.SysDaemon	
* rdrb			
* rdrc			
* rdrd			

The devices are grouped according to the request types they service. An asterisk (\*) before a device indicates that the associated request type is the default for the device. The driver access name and authorization (if any) are given for each request type.

3. To find out what devices and request types (and device classes, if your site uses them) are already running, type:

```
r <l/0 daemon coordinator label> list
```

for example:

```
r cord list
```

4. The system will respond with something like the following:

```
device prta is active, request type printer, request 40017
device prtb is active, request type unlined.confidential,
  request 20007
device rem1 is active, request type rem1_print_unlined,
  request 50002
device rem2.elite is active, request type printer, request 70003
device rem2.pica is active, request type rem2_pica_unlined,
  request 30015
```

## TO GET INFORMATION ABOUT COMMUNICATIONS LINES

1. If you aren't sure what communications lines are available, you can list the line ids by typing:

```
r <1/0 daemon coordinator label> print_line_ids
```

for example:

```
r cord print_line_ids
```

2. The system will respond with:

Line ID	Channel
16	a.h001
2780_1	g.h004

## PROBLEMS

1. If you try to log in a remote driver when you haven't logged in the coordinator, the remote driver waits for you to log in the coordinator. If you don't log in the coordinator within five minutes, the remote driver sends you an error message and logs itself out.
2. The first thing a remote driver tries to do is initialize itself. If it encounters a problem which isn't serious, the system will respond with an error message and the remote driver will ask you for instructions. If it encounters a problem which is serious, the system will respond with an error message that starts with the words "fatal error:" and the remote driver may log out. If you get a fatal error like this, you should bring it to the attention of your system maintainer.
3. If there's some reason why the remote driver is not allowed to operate the device(s) or the request type(s) (or the device class(es), if your site uses them) you specify, the system will respond with one or more error messages explaining the problem, then ask you again for a device name and optional request type.



## SECTION 25

# MANAGING REMOTE DEVICES AT A REMOTE SITE

### TO START UP A REMOTE STATION

This procedure varies quite a bit from site to site. If you have any questions about the right way to do things at your site, ask your system administrator for help.

1. Turn on your terminal. Make sure that the communications lines are connected to the modem and that the terminal is set up to receive data from Multics.
2. If you have a hardwired line, continue with step 4. If you don't, continue with step 3.
3. Dial the central site phone number. Then continue with step 4.
4. If you have a Multics banner on your terminal, continue with step 5. If you don't, continue with step 6.
5. Type:

```
dial <dial id>
```

for example:

```
dial boston_prt
```

Then continue with step 6.

6. If the system responds with:

```
Enter station command:
```

Continue with step 7. If it doesn't, continue with step 10.

7. Type:

```
station <station id>
```

for example:

```
station rem4
```

or, depending on your site's policy:

```
station <station id> <station password>
```

for example:

```
station rem4 secret
```

where "station id" is always the name of the major device. Then continue with step 8.

8. If the system responds with:

```
Enter station password:
```

continue with step 9. If it doesn't, continue with step 10.

9. Type:

```
<station password>
```

for example:

```
secret
```

Then continue with step 10.

10. If the system responds with:

```
Enter request type for minor device <name>:
```

for example:

```
Enter request type for minor device printer:
```

continue with step 11. If it doesn't, continue with step 12.

11. Type:

```
<minor device request type>
```

for example:

```
rem4_print_unlined
```

Repeat steps 10 and 11 for each minor device. Then continue with step 12.

12. If the system responds with messages like these:

For the invoices, use VFU tape number 12.  
The form stock is in storage bins 22, 23 and 24.

Mount VFU tape for 72 lines per page.  
Set printer for 8 lines/inch.

Continue with step 13. If it doesn't, continue with step 14.

13. Take whatever action is indicated by the messages. Then continue with step 14.

14. The system will respond with:

rem4 driver on channel a.h001 ready at 06/01/83 1165.2 est Wed  
Enter command:

15. To start processing requests, type:

go

16. The system will respond with:

Ready (MacKenzie)



PART VII  
EVERYDAY OPERATIONS -- BACKUP



## SECTION 26

# MANAGING BACKUP DAEMONS -- DUMPS

An *incremental dump* locates and dumps all segments and directories which have been created or modified since the last time an incremental dump was done. Its main function is to limit the amount of information that can be lost due to changes that have been made since the last incremental tape was created. An incremental dump is usually run once every hour. It is also called a *backup dump* or a *wakeup dump* (wakeup because it's usually set to "wakeup" by itself at specified intervals).

A *consolidated dump* locates and dumps all segments and directories which have been modified after some specified time in the past. Its main function is to consolidate the most recent information stored on a group of incremental tapes, thus decreasing the number of tapes which must be saved and processed. It does this by dumping all segments which have been modified since the last consolidated dump. A consolidated dump is usually run at the end of every day or every week. It is also called a *catchup dump*.

A *complete dump* locates and dumps every segment and directory in the storage system without regard for when they were last modified. Its main function is to establish a checkpoint in time. If it's ever necessary to recover a major portion of online storage, the tape with the most recent complete dump on it marks a cutoff point, beyond which no older dump tapes are needed. A complete dump is usually run at the end of every week or every month.

*Incremental tapes* (also called *wakeup tapes* or *backup tapes*) are usually kept for about three weeks. *Consolidated tapes* (also called *catchup tapes*) are usually kept for the same amount of time as incremental tapes. *Complete dump tapes* are usually kept for about six months, with the exception of one complete dump tape per month, which is usually kept for a much longer period, say a year or so. The amount of time that these tapes are kept will vary from site to site, as will the scheduling of the various dumps.

Your site may require that hierarchy dump tapes be written at a special density. If this is so, whenever you supply the hierarchy backup system with a tape name, you should also supply it with the density. Instructions for doing this are supplied with all of the examples in this section.

## TO DO AN INCREMENTAL HIERARCHY DUMP

If the hierarchy backup daemon was logged in by the system\_start\_up.ec when you booted Multics, skip to step 3.

1. Type:

```
login Backup.SysDaemon <backup daemon label>
```

for example:

```
login Backup.SysDaemon bk
```

where "bk" is your site's label for the backup daemon that controls hierarchy incremental and consolidated dumps.

2. The system will respond with:

```
as LOGIN Backup.SysDaemon dmn bk (create)
bk r 0940 2.267 34.136 503
bk
->bk
```

3. Type:

```
x inc <your initials> <tape name>
```

or:

```
x inc <your initials> <tape name,d=density>
```

for example:

```
x inc klm ih001
```

or:

```
x inc klm ih001,d=6250
```

where "ih001" is the name of the first tape to be used, and "6250" is the required density.

4. The system will respond with:

```
sc_command r bk start_dump sys_dirs klm l 60
sc_command r bk ih001
bk r 0941 0.324 19.431 152
RCP: Attached tapa_01 for Backup.SysDaemon.z
RCP: Note (tapa_01) - ih001,sys
RCP: Unloading volume from device tapa_01
RCP: Mount Reel ih001 with ring on tapa_01 for
Backup.SysDaemon.z
bk
->bk
```

5. Mount the specified tape on the specified drive. (See "To Mount a Tape" in Section 19.)
6. If another tape is needed, the hierarchy backup daemon will ask you to enter its name. Type:

```
r bk <tape name>
```

or:

```
r bk <tape name,d=density>
```

for example:

```
r bk ih002
```

or:

```
r bk ih002,d=6250
```

7. Repeat steps 4-6 until the daemon is finished doing the dump.
8. Be sure to record the names of the tapes.

#### TO DO AN INCREMENTAL VOLUME DUMP

1. Type:

```
x vinc <your initials>
```

for example:

```
x vinc klm
```

where "vinc" is the short form of the incremental\_volume command.

2. The system will respond with:

```
as LOGIN Volume_Dumper.Daemon dmn vinc (create)
vinc r 0940 2.134 27.369 678
vinc
->vinc
vinc Mounting tape iv001 for writing
RCP: Mount Reel iv001 with ring on tapa_01 for
Volume_Dumper.Daemon
```

3. Mount the specified tape on the specified drive. (See "To Mount a Tape" in Section 19.)

## TO DO A CONSOLIDATED HIERARCHY DUMP

If the hierarchy backup daemon was logged in by the `system_start_up.ec` when you booted Multics, skip to step 3.

1. Type:

```
login Backup.SysDaemon <backup daemon label>
```

for example:

```
login Backup.SysDaemon bk
```

where "bk" is your site's label for the backup daemon that controls hierarchy incremental and consolidated dumps.

2. The system will respond with:

```
as LOGIN Backup.SysDaemon dmn bk (create)
bk r 0940 2.267 34.136 503
bk
->bk
```

3. Type:

```
x cat <your initials> <tape name> <start date> <start time>
```

or:

```
x cat <your initials> <tape name,d=density> <start date>
<start time>
```

for example:

```
x cat klm ch001 05/13/83 1930.0
```

or:

```
x cat klm ch001,d=6250 05/13/83 1930.0
```

where "ch001" is the name of the first tape to be used, "6250" is the required density, and "05/13/83 1930.0" specifies that all segments modified since 05/13/83 at 19:30 be dumped.

4. The system will respond with:

```
bk
bk Type primary dump tape label:
->bk
bk Mounting tape ch001 for writing
RCP: Mount Reel ch001 with ring on tapa_01 for
Backup.SysDaemon
bk
-> bk
```

5. Mount the specified tape on the specified drive. (See "To Mount a Tape" in Section 19.)
6. If another tape is needed, the hierarchy backup daemon will ask you to enter its name. Type:
 

```
    r bk <tape name>
```

 or:
 

```
    r bk <tape name,d=density>
```

 for example:
 

```
    r bk ch002
```

 or:
 

```
    r bk ch002,d=6250
```
7. Repeat steps 4-6 until the daemon is finished doing the dump.
8. Be sure to record the names of the tapes.

#### TO DO A CONSOLIDATED VOLUME DUMP

1. Type:
 

```
    x vcons <your initials>
```

 for example:
 

```
    x vcons klm
```

 where "vcons" is the short form of the consolidated\_volume command.
2. The system will respond with:
 

```
as LOGIN Volume_Dumper.Daemon dmn vcons (create)
vcons r 1719 2.134 27.369 678
vcons
->vcons
vcons Mounting tape cv001 for writing
RCP: Mount Reel cv001 with ring on tapa_01 for
Volume_Dumper.Daemon
```

 where "vcons" is your site's label for the backup daemon that controls volume catchup dumps.
3. Mount the specified tape on the specified drive. (See "To Mount a Tape" in Section 19.)

## TO DO A COMPLETE HIERARCHY DUMP

1. Type:

```
login Dumper.SysDaemon <backup daemon label>
```

for example:

```
login Dumper.SysDaemon cd
```

where "cd" is your site's label for the backup daemon that controls hierarchy complete dumps.

2. The system will respond with:

```
as LOGIN Dumper.SysDaemon dmn cd (create)
cd r 0133 8.765 42.678 369
cd
->cd
```

3. Type:

```
x comp <your initials> <dump control file> <tape name>
```

or:

```
x comp <your initials> <dump control file>
      <tape name,d=density>
```

for example:

```
x comp klm complete hd001
```

or:

```
x comp klm complete hd001,d=6250
```

where "complete" is the name of your site's dump control file (actually "complete.dump"), "hd001" is the name of the first tape to be used, and "6250" is the required density.

- \* 4. The system will respond with:

```
cd
cd Input tape label:
->cd
cd Mounting tape hd001 for writing
RCP: Mount Reel hd001 with ring on tapa_01 for
    Dumper.SysDaemon
cd
-> cd
```

5. Mount the specified tape on the specified drive. (See "To Mount a Tape" in Section 19.)

6. If another tape is needed, the complete hierarchy backup daemon will ask you to enter its name. Type:

```
r cd <tape name>
```

or:

```
r cd <tape name,d=density>
```

for example:

```
r cd hd002
```

or:

```
r cd hd002,d=6250
```

7. Repeat steps 4-6 until the daemon is finished doing the dump.
8. Be sure to record the names of the tapes.

#### TO DO A COMPLETE VOLUME DUMP

1. Type:

```
x vcomp <your initials>
```

for example:

```
x vcomp klm
```

where "vcomp" is the short form of the complete\_volume command.

2. The system will respond with:

```
as LOGIN Volume_Dumper.Daemon dmn vcomp (create)
vcomp r 0133 8.765 42.678 369
vcomp
->vcomp
vcomp Mounting tape vd001 for writing
RCP: Mount Reel vd001 with ring on tapa_01 for
Volume_Dumper.Daemon
```

where "vcomp" is your site's label for the backup daemon that controls volume complete dumps.

3. Mount the specified tape on the specified drive. (See "To Mount a Tape" in Section 19.)

## TO LOGOUT A DUMPER DAEMON

1. If one of the dumper daemons seems to be in trouble, you can log it out.  
Type:

```
logout <backup daemon User_id> <backup daemon label>
```

for example:

```
logout Backup.SysDaemon bk
```

where "bk" is your site's label for the backup daemon that controls hierarchy incremental and consolidated dumps,

or:

```
logout Volume_Dumper.Daemon vinc
```

where "vinc" is your site's label for the backup daemon that controls volume incremental dumps,

or:

```
logout Volume_Dumper.Daemon vcons
```

where "vcons" is your site's label for the backup daemon that controls volume consolidated dumps,

or:

```
logout Dumper.SysDaemon cd
```

where "cd" is your site's label for the backup daemon that controls hierarchy complete dumps,

or:

```
logout Volume_Dumper.Daemon vcomp
```

where "vcomp" is your site's label for the backup daemon that controls volume complete dumps.



2. The system will respond with:

as LOGOUT Backup.SysDaemon dmn bk 0:14 \$1.27 (logout) |

or:

as LOGOUT Volume\_Dumper.Daemon dmn vinc 0:13 \$1.17 (logout) |

or:

as LOGOUT Volume\_Dumper.Daemon dmn vcons 0:12 \$1.07 (logout) |

or:

as LOGOUT Dumper.SysDaemon dmn cd 0:11 \$0.97 (logout) |

or:

as LOGOUT Volume\_Dumper.Daemon dmn vcomp 0:10 \$0.87 (logout) |

#### GOOD IDEAS

1. Hierarchy dump maps are used to locate segments and directories that need to be retrieved. They should be filed in a log, in reverse chronological order, and kept as long as the corresponding tapes are kept.



## SECTION 27

# MANAGING BACKUP DAEMONS -- RETRIEVALS

### TO DO A VOLUME RETRIEVAL

If a user discovers that one of his segments or directories has been lost or damaged, he can request that it be retrieved. He does this by using the "enter\_retrieval\_request" command. His request is queued, and when you do a volume retrieval, it is processed. The system doesn't give you any indication that a user has entered a retrieval request, so you should do volume retrievals at regular intervals.

If the user's segment or directory was deleted from the system more than two weeks ago, his request for a volume retrieval may not be successful. If this is the case, he will have to follow your site's procedure for requesting a hierarchy retrieval.

1. Type:

```
login Volume_Retriever.Daemon <retrieval daemon label>
```

for example:

```
login Volume_Retriever.Daemon vretv
```

where "vretv" is your site's label for the daemon that controls volume retrievals.

2. The system will respond with:

```
as LOGIN Volume_Retriever.Daemon dmn vretv (create)
vretv r 1224 3.987 22.56 123
vretv
->vretv
```

3. Type:

```
r vretv retrieve_from_volume
```

4. As each queued retrieval request is processed, the system will respond with:

```
Recovered segment lost_segment
```

or:

```
Recovered directory damaged_dir
```

where "lost\_segment" and "damaged\_dir" are the names of a retrieved segment and a retrieved directory.

## TO DO A HIERARCHY RETRIEVAL

You should only do a hierarchy retrieval for a user when he or she has followed the correct site procedure for requesting this kind of retrieval. You may also have to do hierarchy retrievals for your system maintainer. He or she may ask you to do them when the volume retriever isn't working or when very large amounts of information need to be retrieved.

For this procedure, you'll need a regular user terminal in addition to an initializer terminal (or the bootload console).

1. At the initializer terminal, type:

```
login Retriever.SysDaemon <retrieval daemon label>
```

for example:

```
login Retriever.SysDaemon r
```

where "r" is your site's label for the daemon that controls hierarchy retrievals.

2. The system will respond with:

```
as LOGIN Retriever.SysDaemon dmn rt (create)
rt r 1248 4.789 33.65 456
rt
->rt
```

3. At the regular user terminal, log in as a regular user by typing:

```
l <your Person_id>
```

for example:

```
l MacKenzie
```

where "l" is the short form of the login command.

4. The system will respond with:

```
Password:
```

5. Type your password. The system will respond with:

```
MacKenzie.Operator logged in 06/01/83 0921.4 est Wed
  from VIP7801 terminal "none."
Last login 03/18/83 0726.2 est Fri from VIP7801
  terminal "none".
r 09:22 3.0 3.50
```

where "r 09:22 3.0 3.50" is a ready message.

6. Type:

```
.cwd >udd>sd>backup
```

where "cwd" is the short form of the change\_wdir command.

7. The system will respond with:

```
r 09:23 2.0 4.90
```

8. Type:

```
qx
```

where "qx" is the short form of the qedx command.

9. There will be no response from the system.

10. Type a list of pathnames, such as these:

```
>udd>ProjectA>User1>Segx
>udd>ProjectB>User4>**
>udd>ProjectA>User2>Segz=>udd>ProjectB>User5>Segz
>udd>ProjectB>User6>**=>udd>ProjectA>User3
```

This will set up a *control file*, which is a segment containing a list of the pathnames of the segments and directories to be retrieved. In this case, the first line of the control file will cause one segment to be retrieved as found. The second line will cause one directory with its entire subtree to be retrieved as found; every segment beginning with >udd>ProjectB>User4 will be retrieved. The third line will cause one segment to be retrieved in a new location; i.e., with the new pathname >udd>ProjectB>User4>Segz. The fourth line will cause one directory with its entire subtree to be retrieved in a new location; every segment beginning with >udd>ProjectB>User6 will be retrieved with a new pathname beginning with >udd>ProjectA>User3. Retrieving a segment with a different pathname is called *doing a cross retrieval*.

11. Again, there will be no response from the system.

12. Type:

```
\f
w <tape name>
q
```

for example:

```
\f
w rt5678
q
```

where "rt5678" is the name of the backup tape you're using for this retrieval.

13. The system will respond with:

```
r 09:26 4.0 4.60
```

14. At the initializer terminal, type:

```
r rt retrieve >udd>sd>backup>rt5678
```

15. The system will respond with:

```
Input tape label
```

16. Type:

```
r rt rt5678
```

17. The system will respond with:

```
RCP: Mount Reel rt5678 with ring on tapa_02
for Retriever.SysDaemon
```

18. Mount the tape on the specified drive. (See "To Mount a Tape" in Section 19.)

19. The tape will spin, and the system will ask you if there are any more tapes.

20. Type:

```
r rt no
```

21. The system will respond with:

```
All requests were satisfied
```

22. At the regular user terminal, log out as a regular user by typing:

```
logout
```

23. The system will respond with:

```
Mackenzie.Operator logged out 06/01/83 0928.3 est Wed
CPU usage 13 sec, memory usage 1.8 units, cost $0.28
hangup
```

## TO LOGOUT A RETRIEVAL DAEMON

1. If one of the retrieval daemons seems to be in trouble, you can log it out.  
Type:

```
logout Volume_Retriever.Daemon <retrieval daemon label>
```

for example:

```
logout Volume_Retriever.Daemon vretv
```

where "vretv" is your site's label for the daemon that controls volume retrievals, or:

```
logout Retriever.SysDaemon <retrieval daemon label>
```

for example:

```
logout Retriever.SysDaemon rt
```

where "rt" is your site's label for the daemon that controls hierarchy retrievals.

2. The system will respond with:

```
as LOGOUT Volume_Retriever.Daemon dmn vretv 0:14 $1.27
(logout)
```

or:

```
as LOGOUT Retriever.SysDaemon dmn rt 0:31 $2.69 (logout)
```

## PROBLEMS

1. If you encounter problems with some of the requests when you're doing a volume retrieval, try typing:

```
r vretv retrieve_from_volume -step -long
```

2. Each retrieval request will be printed, along with this prompt:

Command:

3. You should respond with one of the following commands:

proceed, p	process the retrieval request and delete it from the queue.
cancel, c	do not process the retrieval request; delete it from the queue.
skip, s	do not process the retrieval request, but leave it in the queue.

### GOOD IDEAS

1. You may want to defer retrievals until a time when the system load isn't at a peak (say, late afternoon or evening). A lot of tape activity puts a heavy load on the system and adversely affects users.



**PART VIII**  
**EVERYDAY OPERATIONS -- MISCELLANEOUS**

## SECTION 28

# COMMUNICATING WITH USERS

### TO SEND A MESSAGE TO ALL USERS AT LOGIN

When you send a message to all users at login, it means that every time any user logs in, he or she receives the message.

1. Type:

```
word login <"message">
```

for example:

```
word login "Only one CPU until 1300."
```

2. The system will respond with:

```
as word: normal message: Only one CPU until 1300.
```

### TO SEND A MESSAGE TO ONE USER

1. Type:

```
w <user name> <"message">
```

for example:

```
w User4.ProjectB "Can't find your tape. Call X7739."
```

where "w" is the short form of the warn command, "User4.ProjectB" is the name of the user to whom you want to send the message, and "Can't find your tape. Call X7739." is the message you want to send.

2. The system will respond with:

```
Ready (MacKenzie)
```

You should not use the warn command for casual communication, since the message is forced into the middle of anything the user is typing, ruining his or her output. Use it only when you have an urgent message for a particular user who can't be reached by telephone.

## TO SEND A MESSAGE TO ALL USERS

1. Type:

```
w * * <"message">
```

for example:

```
w * * "The system is about to crash."
```

where "w" is the short form of the warn command.

2. The system will respond with:

```
Ready (MacKenzie)
```

As mentioned above, the warn command should not be used for casual communication. Use it only when the system is about to crash or when you are doing a nonscheduled shutdown.

## PROBLEMS

1. If you try to send a message to a user who specified the `-no_warning` argument at login, the system will respond with the following message:

```
warn: User4.ProjectB has "no_warning"
```

If the system can't find the user, you'll also get an error message. If either of these things happens, you'll have to contact the user by some other means.

## SECTION 29

# MANAGING USER LOGINS, LOGOUTS, AND CHANNELS

### TO FIND OUT HOW MANY USERS ARE LOGGED IN

1. Type:

hmu

2. The system will respond with:

```
Multics 11.0: Yoursite, Yourtown, Yourstate
Load = 41.0 out of 110.0 units: users = 41
Absentee users = 3; Maximum absentee users = 10
```

### TO FIND OUT WHICH USERS ARE LOGGED IN

1. Type:

who

2. The system will respond with:

	Login at	TTY	Load	Chan	Group	PNDS	User ID
	05/31/83	1206.8	none	1.0	c.h118	System	MacKenzie.Operator
	06/01/83	0957.3	H19	1.0	a.h012	SysProg > DS	Bongo.Multics
		1007.9	none	1.0	c.h003	Fourth S	User4.ProjectB
		1020.6	none	1.0	a.h022	SysAdm +N	Sissy.SysAdmin
		1021.2	none	0.5	a.h100		*Student43.Class
		1132.7	Q 3	1.0	abs1	Third A	User1.ProjectA
		1133.6	cord	0.5	cord	10 D	10.SysDaemon
		1134.5	prta	0.5	prta	10 D	10.SysDaemon
		2030.3	bk	1.5	bk	System D	Backup.SysDaemon
		2031.2	vinc	1.5	vinc	System D	Volume_Dumper.Daemon

### TO SET THE MAXIMUM NUMBER OF USERS WHO CAN LOG IN

It's a good idea to perform this procedure after you've added or deleted processors or memory, because different configurations can support different numbers of users.

1. To set the maximum number of users to the automatic value set by your system administrator, type:

```
maxu auto
```

To set the maximum number of users to some other value, type:

```
maxu <10X load units>
```

for example:

```
maxu 413
```

where "maxu" is the short form of the maxunits command, and "413" is ten times the maximum number of load units. (In other words, 413 sets the maximum number of load units to 41.3.) Your site administrator will provide you with the correct value.

2. The system will respond with:  

```
as maxu: 413 (41 normal users)
```
3. If you set the maxu value to a very small number, the system will send you a warning:  

```
as warning: maxunits are in tenths of load units.  
as You just set maxu to 20, or 2 normal users.  
as maxu: 20 (2 normal users)
```
4. If you've made a mistake, just retype the maxu command with the correct maxu value.

## TO LOG A USER OUT

You should only log a user out at his or her request or at the request of your system administrator.

1. Type:

```
bump <user name> <time> <"message">
```

for example:

```
bump User6.ProjectA 3 "Bumping you as you requested."
```

or:

```
bump <terminal channel> <time> <"message">
```

for example:

```
bump a.h013 3 "Bumping you as you requested."
```

where "User6.ProjectA" is the name of the user, "a.h013" is the channel to which the user's terminal is connected, "3" is the number of minutes the user has before the bump takes place, and "Bumping you as you requested" is a message of your choice. If you must log out the user immediately, don't type a time.

2. The system will respond with:

```
bump: User6.ProjectA bumped
```

3. Sometimes, even though you get this message, the user is not logged out. In this case, try detaching the channel, which should disconnect the user's terminal. Type:

```
detach a.h013
```

4. The system will respond with:

```
as detach: a.h013 detached
```

5. The user should be logged out immediately. If he isn't, try removing the channel all together. Type:

```
remove a.h013
```

6. The system will respond with:

```
as remove: a.h013 removed
```

7. If you are forced to remove a channel in order to log a user out, you should attach the channel right after, which will make it available for use again. Type:

```
attach a.h013
```

8. The system will respond with:

```
as attach: a.h013 attached
```

## TO RESPOND TO A USER WHO'S BEEN LOGGED OUT

If a user has been logged out by the system, you may get an angry phone call. If you can explain to the user exactly why he's been logged out, you may be able to calm him down. Recall the discussion of load control groups in Section 3. Explain to the user that if a secondary user is logged in and a primary user wants to log in, the secondary user may be preempted by the primary user if system resources are limited. If necessary, refer the user to your system administrator.

## PROBLEMS

1. If the FNP masks a channel, you will get the following message:

```
dn355: FNP masked channel a.h022 for excessive interrupts.
```

In most cases, you should consult with your system administrator. The exception is when a user calls and asks you to restart his or her channel. In that case, type:

```
attach <terminal channel>
```

for example:

```
attach a.h022
```

where "a.h022" is the terminal channel mentioned in the error message. If the user is still having problems, don't type the command again without talking to your system administrator first.

## SECTION 30

# MANAGING ABSENTEE JOBS

At some sites, the majority of the tasks described in this section are performed by the system administrator. If this is the case at your site, you shouldn't attempt to perform these tasks yourself without consulting your system administrator first.

### TO START THE ABSENTEE FACILITY

At most sites, the absentee facility is started when the answering service is started (i.e., when you boot Multics). But if this isn't the case at your site, or if you've stopped the absentee facility for some reason, you can start it yourself.

1. To start the absentee facility with the automatic values set by your system administrator, type:

```
abs start
```

To start the absentee facility with other values, type:

```
abs start <maximum jobs> <queue number>
```

for example:

```
abs start 6 4
```

where "6" is the maximum number of absentee jobs that can run simultaneously, and "4" is the highest numbered (lowest priority) queue.

2. The system will respond with:

```
Ready (MacKenzie)
```

### TO STOP THE ABSENTEE FACILITY

*Note:* the procedure below for doing an "abs stop now" bumps absentee users *immediately*, before they have a chance to do anything. Obviously, you should only perform this procedure when absolutely necessary.

1. To stop the absentee facility in 20 minutes, type:

```
abs stop
```

2. In 20 minutes, the system will respond with:

```
as abs: All absentee processes have run to completion.
```



3. To stop the absentee facility immediately, type:

```
abs stop now
```

4. The system will respond with:

```
abs: Bumping all remaining absentee processes.
```

then with:

```
as abs: All absentee processes have run to completion.
```

### TO SET THE NUMBER OF ABSENTEE SLOTS

The number of absentee slots is the maximum number of absentee jobs that can run simultaneously.

1. To set the number of slots to its automatic value, type:

```
abs maxu auto
```

To set the number of slots to another value, type:

```
abs maxu <maximum jobs>
```

for example:

```
abs maxu 6
```

where "6" is the maximum number of jobs that can run simultaneously.

2. The system will respond with:

```
Ready (MacKenzie)
```

3. To check the value, type:

```
hmu
```

4. The system will respond with:

```
Multics 11.0: Yoursite, Yourtown, Yourstate  
Load = 41.0 out of 110.0 units: users = 41  
Absentee users = 2; Maximum absentee users = 6
```

### TO STOP AN ABSENTEE QUEUE

If you're having problems with a queue and the system doesn't drop it, you can stop it yourself.

1. Type:

```
abs stop queue <queue number>
```

for example:

```
abs stop queue 3
```

2. The system will respond with:

```
as
Queue errors  (*=dropped)
as  FG  0  1  2  3  4
as  0  0  0  0  0  0
as                                *
```

## TO RESTART AN ABSENTEE QUEUE

If a queue has been dropped by the system due to errors, or if you've stopped it yourself, you can restart it yourself.

1. Type:

```
abs start queue <queue number>
```

for example:

```
abs start queue 2
```

2. The system will respond with:

```
Ready (MacKenzie)
```

## TO MOVE AN ABSENTEE JOB FROM ONE QUEUE TO ANOTHER

Multics absentee jobs are routed to different queues, which have high or low priorities assigned to them. Queue 1 has the highest priority; queue 4 has the lowest. Normally, user jobs are routed to queue 3. Your system administrator will usually put system maintenance jobs in queue 1 or queue 2. If given the correct access by the system administrator, users may also put jobs in these higher priority queues. If necessary, you can move a job from one queue to another.

1. Type:

```
abs move -user <user name> -id <id number>
        -to_q <queue number>
```

for example:

```
abs move -user User6.ProjectB -id 175517.0 -to_q 1
```

where "User6.ProjectB" is the name of the user to whom the job belongs, "175517.0" is the id number displayed by the abs list command (see "To Get Information About Absentee Jobs" later in this section), "-to\_q" is short for -to\_queue, and "1" is the queue you want to move the job to.

2. The system will respond with:

```
Ready (MacKenzie)
```

## TO SET THE HIGHEST NUMBERED QUEUE WHICH WILL BE SERVICED

The highest numbered queue which will be serviced is the lowest priority queue.

1. To set the highest numbered queue to its automatic value, type:

```
abs maxq auto
```

To set the highest numbered queue to another value, type:

```
abs maxq <queue number>
```

for example:

```
abs maxq 4
```

2. The system will respond with:

```
Ready (MacKenzie)
```

3. To check the value, type:

```
abs maxq
```

4. The system will respond with:

```
as abs maxq (manual): 4
```

## TO BUMP AN ABSENTEE JOB

Bumping a job means stopping it when it's in the middle of running. A bumped job which is restartable remains in the queue and is restarted at a later time.

1. Type:

```
abs bump <job name>
```

for example:

```
abs bump >udd>ProjectB>User5>list.absin
```

or:

```
abs bump abs<slot number>
```

for example:

```
abs bump abs1
```

where "1" is the number of the slot in which the job is running. To find out the slot number, use the who command as described in Section 29 and look under the "chan" heading of the output.

2. The system will respond with:

```
as LOGOUT User5.ProjectB Q 3 abs1 0:20
$0.05 (bump)
```

## TO DEFER AN ABSENTEE JOB

You should defer a job if it's not supposed to run until a particular event happens. An example of this is a job that's not supposed to run until a tape reel is delivered to the machine room. A deferred job remains deferred until it is released. A deferred job is preserved across shutdowns and crashes. You may release it whenever it's appropriate to do so.

1. Type:

```
abs defer <job name>
```

for example:

```
abs defer >udd>ProjectA>User2>short.absin
```

2. The system will respond with:

```
abs defer: 1 request deferred.
```

## TO SUSPEND AN ABSENTEE JOB

You may suspend a job (or jobs) when you need to reduce the load on the system. A suspended job remains suspended until it is released. A suspended job is *not* preserved across shutdowns and crashes. You should release it as soon as possible.

1. Type:

```
abs suspend <job name>
```

for example:

```
abs suspend >udd>ProjectB>User4>long.absin
```

2. The system will respond with:

```
abs suspend: 1 request suspended.
```

## TO RELEASE AN ABSENTEE JOB

When a job is released, it's removed from the deferred or suspended state, and allowed to log in or resume running.

1. Type:

```
abs release <job name>
```

for example:

```
abs release >udd>ProjectA>User3>short.absin
```

2. The system will respond with:

```
abs release: 1 request released.
```

## TO FORCE A JOB TO LOG IN NEXT

When you force a job to log in next, you make it run next without being subject to any of the usual limitations. This may create problems, so you should only do it when absolutely necessary.

1. Type:

```
abs run <job name> -user <user name>
```

for example:

```
abs run >udd>ProjectB>User6>now.absin -user User6.ProjectB
```

where ">udd>ProjectB>User6>now.absin" is the name of the job, and "User6.ProjectB" the name of the user to whom the job belongs.

2. The system will respond with:

```
as LOGIN User6.ProjectB Q 3 abs1 (create) [daily]
```

3. If necessary, the system will warn you that logging in the job next may create problems. It will ask you if the job should be logged in anyway, despite the potential difficulties. You should answer "yes" or "no".

## TO GET INFORMATION ABOUT ABSENTEE JOBS

1. To list all absentee jobs, type:

```
abs list -et **
```

where "-et" is short for -entry. (If you don't want to list jobs which have been deferred, type: "abs list -im -et \*\*.")

2. The system will respond with:

```
Absentee queue 3: 2 requests. 2 total requests  
(1 deferred).
```

User	ID	Input Segment
Bongo.SysMaint	175517.0	ear.absin (running)
User1.ProjectA	170063.2	carry.absin (deferred by user to 06/02/83 1300.0)

3. To list all deferred absentee jobs, type:

```
abs list -et ** -dfi
```

where "-et" is short for -entry, and "-dfi" is short for -deferred\_indefinitely.

4. The system will respond with:

```
Absentee queue 3: 1 request. 2 total requests  
(1 deferred).
```

User	ID	Input Segment
User1.ProjectA	170063.2	carry.absin (deferred indefinitely by user)

5. To list a job's position in the queue, type:

```
abs list <job name> -psn
```

for example:

```
abs list >udd>ProjectB>User4>list.absin -psn
```

where "-psn" is short for -position.

6. The system will respond with:

```
Absentee queue 3: 1 request. 2 total requests  
(1 deferred).
```

User	ID	Input Segment
Bongo.SysMaint	(2) 175517.0	ear.absin

7. To list the total number of jobs in each queue, type:

```
abs list -et ** -tt
```

where "-et" is short for -entry, and "-tt" is short for -total.

8. The system will respond with:

```
Absentee queue 1: 10 requests. 10 total requests  
(10 deferred).  
Absentee queue 3: 2 requests. 2 total requests  
(1 deferred).
```

## SECTION 31

# SETTING ATTENDED/UNATTENDED MODE AND MANUAL/AUTOMATIC MODE

Normally, the system runs in *attended mode* (also called *attended service*). In attended mode, the system assumes that you are present to mount tapes and disks. However, if you need to be away from the machine room for an extended period of time, or if the system is going to run during a night or a weekend when no operator will be present, you can arrange to have the system run in *unattended mode* (also called *unattended service*). In unattended mode, it appears to RCP that all tape drives are deleted and no I/O daemons are available. This means that user tape mount requests are not honored, and user print requests line up in the queues. If backup is running, you must preload tapes for it, because you won't be available to mount them at the time they're needed. See "To Preload a Tape" in Section 19. The greeting message is set to "Unattended service."

Normally, the system runs in *manual mode*. In manual mode, the system pauses in BCE after a crash, and waits for you to take action. However, if you need to be away from the machine room for an extended period of time, or if the system is going to run during a time when no operator will be present, you can arrange to have the system run in *automatic mode*. In automatic mode (also called *automatic reboot mode*), the system automatically takes a dump and performs an ESD after a crash. (If the system is in unattended mode, it may also reboot itself automatically). Remember that if you use the auto exec\_com to boot Multics, automatic mode gets turned on automatically. Since manual mode is the usual way of running things, if you use the auto exec\_com to boot Multics, you should always turn off automatic mode as soon as the system is up.

It makes sense to put the system in automatic mode whenever you put it in unattended mode, and vice versa.

### TO SET UNATTENDED MODE

1. Type:

```
x unattend
```

where "unattend" is the short form of the unattended command.



2. The system will respond with:

```
Ready (MacKenzie)
as word: normal message: Unattended service.
RCP: Deleted device tapa_01
RCP: Deleted device tapa_02
RCP: Deleted device tapa_03
RCP: Deleted device tapa_04
RCP: Deleted device tapa_05
RCP: Deleted device tapa_06
RCP: Deleted device tapa_07
RCP: Deleted device tapa_08
```

There will be one tape drive deletion message for each of your tape drives.

### TO SET ATTENDED MODE

1. Type:

```
x attend
```

where "attend" is the short form of the attended command.

2. The system will respond with:

```
Ready (MacKenzie)
RCP: Added device tapa_01
RCP: Added device tapa_02
RCP: Added device tapa_03
RCP: Added device tapa_04
RCP: Added device tapa_05
RCP: Added device tapa_06
RCP: Added device tapa_07
RCP: Added device tapa_08
```

There will be one tape drive addition message for each of your tape drives.

### TO SET AUTOMATIC MODE

1. Type:

```
x auto on
```

2. The system will respond with:

```
Ready (MacKenzie)
```

## TO SET MANUAL MODE

1. Type:  
x auto off
2. The system will respond with:  
Ready (MacKenzie)



## SECTION 32

# DOING DYNAMIC RECONFIGURATION

Dynamic reconfiguration is the process of adding or deleting a major hardware module or peripheral to or from the Multics configuration when the system is at Multics level (i.e., while the system is running). You do this by executing commands.

Usually, the reason you delete a box or a peripheral dynamically is because there are problems with it. Usually, the reason you add a box or a peripheral dynamically is because there were problems with it, and CSD has fixed it. You can't add or delete a box or a peripheral for which there was no card in the config deck when the system was booted. When you add or delete a box or a peripheral dynamically, the system changes the card for you automatically.

This section explains the procedures for adding and deleting processors, memory, IOMs, FNPs, and channels, and the procedures for adding an alternate bootload console, deleting the bootload console, and changing the bootload console. Procedures for adding and deleting tape drives are explained in Section 19. Procedures for adding and deleting disk drives are explained in Sections 20 and 21.

### TO ADD A PROCESSOR

1. The first thing you should do is make sure that all of the switches on the processor are set correctly, in the usual way. Refer to Section 9 and Appendix A.
2. Before a processor can be added to the system, it must be initialized. To initialize a Level 68 processor, leave the PORT ENABLE switches on, and press the INITIALIZE & CLEAR button on the configuration panel at least 4 times. Each time you press the INITIALIZE & CLEAR button on the Level 68 processor, wait about 5 seconds for the CPU to finish before you press it again. To initialize a DPS 8 processor, leave the PORT ENABLE switches on, and press the INIT-CLEAR button on the configuration panel at least 4 times. Each time you press the INIT-CLEAR button on the DPS 8 processor, wait for the CPU self tests to finish before you press it again. When the self tests are finished, the green light goes on. This light is located on the right hand side of the MP board (the topmost board in the CPU cabinet), just behind the free-edge connector for that board. Since the light is inside the CPU in an awkward position, it may be easier for you to simply wait 10 to 15 seconds before you press the INIT-CLEAR button again.

3. Type:

```
rcf add cpu <cpu tag>
```

for example:

```
rcf add cpu a
```

where "rcf" is the short form of the reconfigure command.

4. The system will respond with:

```
start_cpu: Added CPU A.
```

then with:

```
reconfigure: CPU a is now running.
```

## TO DELETE A PROCESSOR

1. Type:

```
rcf dl cpu <cpu tag>
```

for example:

```
rcf dl cpu a
```

where "rcf" is the short form of the reconfigure command, and "dl" is short for delete.

2. The system will respond with:

```
stop_cpu: Removed CPU A.
```

then with:

```
reconfigure: Deleted CPU a.
```

3. If you delete the bootload CPU, you will also get a message indicating which CPU the system now considers to be the bootload CPU. Pay special attention to this message. You will need this information if you have to execute fault or execute switches. (See Section 36.) The best idea is to add a note to the system log book recording which CPU is the new bootload CPU.

## TO ADD MEMORY

1. The first thing you should do is make sure that all of the switches on the SCU are set correctly, in the usual way. Refer to Section 9 and Appendix A.
2. Before an SCU can be added to the system, its store units must be cleared.

## Clearing the Store Units

3. Turn all of the PORT ENABLE switches on the configuration panel to OFF.
4. On a L68 SCU, press the MEM CLEAR and SYNDROME RESET buttons on each memory control panel. (There's one panel for each store unit.)

On a DPS 8 SCU, locate the MEMORY SELECT switches on the memory control panel. (The right switch has a "1" above it and the left switch has a "2" above it.) Set the switches to 00 (both switches down) and press the MEM CLEAR and SYNDROME RESET buttons. Then set the switches to 01 (right switch up, left switch down) and press the MEM CLEAR and SYNDROME RESET buttons again. Then set the switches to 10 (right switch down, left switch up) and press the MEM CLEAR and SYNDROME RESET buttons again. Finally, set the switches to 11 (both switches up) and press the MEM CLEAR and SYNDROME RESET buttons again. If any store unit isn't being used, you don't have to select it. CSD can tell you which store units aren't being used.

5. Press the PANEL CLEAR button on the maintenance panel.
6. Press the INITIALIZE button on the maintenance panel.
7. Turn the PORT ENABLE switches on the configuration panel which correspond to the IOM(s) and the bootload CPU to ON.
8. Set the MASK A MASK/PORT ASSIGNMENT switch on the configuration panel to the number of the SCU port to which the bootload CPU is connected. (Note: this is *NOT* the number of the bootload CPU.)
9. Toggle the MODE switch on the configuration panel. (Set it to MANUAL, then set it back to PROGRAM.)

## Adding the Memory

11. Type:

```
rcf add mem <scu tag>
```

for example:

```
rcf add mem c
```

where "rcf" is the short form of the reconfigure command.

12. The system will respond with:

```
addmem: Added MEM C.
```

and:

```
reconfigure: Added SCU c and its memory.
```

## TO DELETE MEMORY

1. Low-order memory can't be deleted — in other words, the bootload SCU can't be removed from the configuration.

2. Type:

```
rcf dl mem <scu tag>
```

for example:

```
rcf dl mem c
```

where "rcf" is the short form of the reconfigure command, and "dl" is short for delete.

3. The system will respond with:

```
delmem: Removed MEM C.
```

and:

```
reconfigure: Removed SCU c and its memory.
```

## TO ADD AN IOM

1. If you are adding an IOM (but not an IMU), the first thing you should do is make sure that all of the switches on the IOM are set correctly, in the usual way. Refer to Section 9 and Appendix A. Then continue with step 3.
2. If you are adding an IMU, the first thing you should do is find out whether you are supposed to use the default IMU configuration file or one of the other configuration files. Make sure that the diskette which contains the configuration file you're supposed to use is in one of the IMU diskette drives. Then continue with step 4.
3. Before an IOM can be added to the system, it must be initialized. To initialize a Level 68 IOM, go to the maintenance panel. Set the IOM INITIALIZE switch to MANUAL and press the MANUAL button. (Do *NOT* confuse this button with the SYSTEM INITIALIZE button on the bootload panel.) To initialize a DPS 8 IOM, go to the configuration panel. Press the INITIALIZE button. (Do *NOT* confuse this button with the SYSTEM INIT button on the bootload panel.) Then continue with step 5.
4. Before an IMU can be added to the system, it must be initialized. To initialize an IMU, use the rload MCA command (rload imu). Do *NOT* use the init MCA command; this will initialize the entire system. Then continue with step 5.

5. Type:

```
rcf add iom <iom tag> -add_all_attachments
```

for example:

```
rcf add iom b -add_all_attachments
```

where "rcf" is the short form of the reconfigure command.

6. The system will respond with:

```
RCF: Added IOM b.
```

then with other messages, depending on which other devices are added.



This page intentionally left blank.

## TO DELETE AN IOM

1. Type:

```
rcf dl iom <iom tag> -delete_all_attachments
```

for example:

```
rcf dl iom b -delete_all_attachments
```

where "rcf" is the short form of the reconfigure command, and "dl" is short for delete.

2. The system will respond with various messages, depending on which other devices are deleted, then with:

```
RCF: Deleted IOM b.
```

## TO ADD AN FNP

1. The first thing you should do is make sure that all of the switches on the FNP are set correctly, in the usual way. Refer to Section 9 and Appendix A.

2. Type:

```
rcf add dv <FNP name>
```

for example:

```
rcf add dv fnpa
```

where "rcf" is the short form of the reconfigure command, "dv" is short for device, and "fnpa" is the name of the FNP you want to add.

3. The system will respond with:

```
RCP: Added device fnpa
```

then with:

```
fnp_util: FNP a added to configuration.
```

4. To load the FNP so that users can log in over its channels, type:

```
load_mpx <FNP tag>
```

for example:

```
load_mpx a
```

5. The system will respond with:

```
FNP a loaded successfully.
```

## TO DELETE AN FNP

1. Type:

```
rcf dl dv <FNP name>
```

for example:

```
rcf dl dv fnpa
```

where "rcf" is the short form of the reconfigure command, "dl" is short for delete, "dv" is short for device, and "fnpa" is the name of the FNP you want to delete.

2. The system will respond with:

```
fnp_util: FNP a deleted from configuration.
```

then with:

```
RCP: Removed device fnpa.
```

*Note:* if users are logged in over the FNP's channels, you will not be able to delete it.

## TO ADD A LOGICAL CHANNEL

1. Type:

```
rcf add chan <channel name>
```

for example:

```
rcf add chan b23
```

where "rcf" is the short form of the reconfigure command, "chan" is short for channel, and "b23" is the name of the logical channel you want to add.

2. The system will respond with:

```
reconfigure: Added logical channel B23.
```

## TO DELETE A LOGICAL CHANNEL

1. Type:

```
rcf dl chan <channel name>
```

for example

```
rcf dl chan b23
```

where "rcf" is the short form of the reconfigure command, "dl" is short for delete, "chan" is short for channel, and "b23" is the name of the logical channel you want to delete.

2. The system will respond with:

```
reconfigure: Removed logical channel B23.
```

## TO ADD AN ALTERNATE BOOTLOAD CONSOLE

*Note:* if operators are not allowed to use the sac command at your site, you will not be able to perform this procedure.

1. At a regular user terminal, log in as a regular user by typing:

```
l <your Person_id>
```

for example:

```
l MacKenzie
```

where "l" is the short form of the login command.

2. The system will respond with:

```
Password:
```

3. Type your password. The system will respond with:

```
MacKenzie.Operator logged in 06/01/83 0921.4 est Wed  
from VIP7801 terminal "none".  
Last login 03/18/83 0726.2 est Fri from VIP7801  
terminal "none".  
r 09:22 3.0 3.50
```

where "r 09:22 3.0 3.50" is a ready message.

4. Type:

```
sac ssc <console name> -state alt
```

for example:

```
sac ssc opcd -state alt
```

where "ssc" is the short form of the set\_system\_console command, and "opcd" is the name of the alternate console as it appears in the config deck. *Note:* use of this command requires that your system administrator give your User\_id special privilege.

5. The system will respond with:

```
r 09:26 4.0 2.60
```

6. Log out as a regular user by typing:

```
logout
```

7. The system will respond with:

```
MacKenzie.Operator logged out 06/01/83 0928.3 est Wed  
CPU usage 13 sec, memory usage 1.8 units, cost $0.28  
hangup
```

## TO CHANGE THE BOOTLOAD CONSOLE

*Note:* if operators are not allowed to use the sac command at your site, you will not be able to perform this procedure.

1. At a regular user terminal, log in as a regular user by typing:

```
l <your Person_id>
```

for example:

```
l MacKenzie
```

where "l" is the short form of the login command.

2. The system will respond with:

```
Password:
```

3. Type your password. The system will respond with:

```
Mackenzie.Operator logged in 06/01/83 0921.4 est Wed
from VIP7801 terminal "none".
Last login 03/18/83 0726.2 est Fri from VIP7801
terminal "none".
r 09:22 3.0 3.50
```

where "r 09:22 3.0 3.50" is a ready message.

4. Type:

```
sac ssc <console name> -state on
```

for example:

```
sac ssc opcb -state on
```

where "ssc" is the short form of the `set_system_console` command, and "opcb" is the name of the new bootload console as it appears in the config deck. *Note:* use of this command requires that your system administrator give your `User_id` special privilege.

5. The system will respond with:

```
Console opcb assigned by Initializer.SysDaemon
The current bootload console will be marked
as an I/O device.
```

6. Log out as a regular user by typing:

```
logout
```

7. The system will respond with:

```
Mackenzie.Operator logged out 06/01/83 0928.3 est Wed
CPU usage 13 sec, memory usage 1.8 units, cost $0.28
hangup
```

## TO DELETE THE BOOTLOAD CONSOLE

*Note:* under certain circumstances, deleting the bootload console will crash the system. For this reason, you should *never* attempt to delete the bootload console without consulting your system maintainer first. If operators are not allowed to use the `sac` command at your site, you will not be able to perform this procedure at all.

1. At a regular user terminal, log in as a regular user by typing:

```
| l <your Person_id>
```

for example:

```
| l MacKenzie
```

where "l" is the short form of the login command.

2. The system will respond with:

```
    Password:
```

3. Type your password. The system will respond with:

```
    MacKenzie.Operator logged in 06/01/83 0921.4 est Wed
      from VIP7801 terminal "none".
    Last login 03/18/83 0726.2 est Fri from VIP7801
      terminal "none".
    r 09:22 3.0 3.50
```

where "r 09:22 3.0 3.50" is a ready message.

4. Type:

```
    sac ssc <console name> -state io
```

for example:

```
    sac ssc opca -state io
```

where "ssc" is the short form of the set\_system\_console command, and "opca" is the name of the deleted console as it appears in the config deck. *Note:* use of this command requires that your system administrator give your User\_id special privilege.

5. The system will respond with:

```
    r 09:26 4.0 2.60
```

6. Log out as a regular user by typing:

```
    logout
```

7. The system will respond with:

```
    MacKenzie.Operator logged out 06/01/83 0928.3 est Wed
    CPU usage 13 sec, memory usage 1.8 units, cost $0.28
    hangup
```

## SECTION 33

# LEAVING A NOTE IN THE SYSTEM LOG BOOK

Anytime something unusual happens, you should make a note of it in the system log book, for future reference. Unusual occurrences include all of the following:

- hardware failures
- serious salvager errors
- all system crashes. (Notes on system crashes should include the apparent state of the system at the time of the crash, the names of any hardware error lights which were on on any CPUs, SCUs, IOMs, or MPCs (for example, CPU parity lights), messages you received on the bootload console at the time of the crash, the method you used to return to BCE, and the type of dump you took.)
- changes to the hardware configuration
- changes to the configuration deck
- changes to normal schedules
- changes to disk packs
- a different CPU used for the bootload
- any deviations from normal procedures

It's also helpful to make notes on the following:

- what the system performance was like during your shift
- when complete dumps were done
- the names of hierarchy dump tapes

### GOOD IDEAS

1. When you start your shift, take a look at the notes in the log book from the previous shift. This will give you an idea of what kind of shape the system is in.





PART IX  
SPECIAL OPERATIONS

# SECTION 34

## USING THE DPU AND THE DMP/VIP

### TO USE THE DPU

#### DPU Typing Conventions

All input to the DPU must be in upper case. The easiest way to accomplish this is to press the CAPS LOCK key and leave it depressed for the remainder of your DPU session.

All input to the DPU, including the use of special controls such as the DEL key and the function keys, must be terminated by typing a carriage return (pressing the RETURN or CR key). This is true no matter what mode the DPU is in. (DPU modes are described later in this section.)

Exceptions to the above rules are the character and line kill keys. They are not typed in upper case, and are not followed by a carriage return. To delete an input character, type "@". To delete an input line, type CTL-X. (This is done by pressing the key marked CTL and keeping it depressed while you type an X.)

#### DPU Naming Conventions

Each major hardware module that is cabled to the DPU has an assigned *keyname* that is used by the DPU to uniquely identify that unit. The keynames are as follows:

CPMxx	DPS 8 CPU
SCUxx	DPS 8 SCU
IOMxx	DPS 8 IOM

where xx is a two-digit number from 00 to 99 that is used to uniquely identify a specific unit in a multi-unit configuration. This number is known as a *tag*.

#### DPU Modes

The DPU can be operated in several different modes. The first mode is *ready mode*, the one you enter after you boot the DPU. Ready mode is indicated by this prompt:

C?

When you're in ready mode, you can change the unit connected to the DPU, and get help with DPU commands.

From ready mode, you can enter a second mode called *transparent* or *TM mode*. TM mode is indicated by this prompt:

OFL?

When you're in TM mode, you can find out which unit is connected to the DPU.

From TM mode, you can enter a third mode called *VIP mode*. VIP mode is indicated by this prompt:

<UNIT> CMD

for example:

CPM CMD

which means that a CPU is connected to the DPU.

When you're in VIP mode, you can display the configuration panels of the CPU, SCU and IOM, execute switches to return to BCE, and put processors in and out of step. (Instructions for executing switches and putting processors in and out of step are given in Section 36.)

#### GETTING INTO TM MODE FROM READY MODE

1. Type:

OFL <KEYNAME>

for example:

OFL CPM00

where "keyname" is the unit connected to the DPU, in this case CPU00.

2. The system will respond with:

RD CMD FILE

then with:

OFL?

*GETTING INTO VIP MODE FROM TM MODE*

1. Type:  
VIP
2. The system will respond with:  
CPM CMD

*GETTING INTO TM MODE FROM VIP MODE*

1. Hit the BRK key, or type:  
TM
2. The system will respond with:  
OFL?

*GETTING INTO READY MODE FROM TM MODE*

1. Hit the BRK key, or type:  
QUIT
2. The system will respond with:  
C?

**Finding Out Which Unit is Connected to the DPU**

1. Get into TM mode.
2. Type:  
.U
3. The system will respond with:  
<KEYNAME>

for example:

CPM00

which means that CPU00 is connected to the DPU, then with:

OFL?

## Changing the Unit Connected to the DPU

1. Get into ready mode.
2. Type:

OFL <KEYNAME>

for example:

OFL SCU01

which means you want to connect SCU01 to the DPU.

3. The system will respond with:

RD CMD FILE

then with:

OFL?

indicating that you are now in TM mode. Note that this procedure is the same as the one you use to get into TM mode from ready mode.

## Getting Help With DPU Commands

1. Get into ready mode.
2. Type:  
?
3. The system will respond with information about available DPU commands and how to use them.

## TO USE THE DMP/VIP

### DMP/VIP Typing and Naming Conventions

Typing conventions on the DMP/VIP are the same as on the DPU, with one exception. On the DMP/VIP, you don't have to type a carriage return after you press a function key.

Naming conventions are exactly the same on the DMP/VIP as on the DPU.

## DMP/VIP Modes

The DMP/VIP operates in just one mode: VIP mode. Being in VIP mode on the DMP/VIP is the same as being in VIP mode on the DPU. The only difference is that you don't have to install the site configuration and go through ready mode and TM mode to get to it.

When you are in VIP mode on the DMP/VIP, you can do all of the things you can do in VIP mode on the DPU. These include displaying configuration panels, executing switches, and putting CPUs in and out of step.

### Getting Connected to the DMP/VIP

1. If each unit (CPU, IOM, SCU) has its own DMP/VIP terminal, you can just type on the terminal connected to the unit you want to work with. Otherwise, you must get the DMP/VIP terminal hooked up to the desired unit before you can begin typing.
2. Enter several carriage returns. This will enable the DMP to determine the baud rate of your terminal. Once the DMP has done this, you will receive the following message:

`<UNIT> CMD`

for example:

`CPM CMD`

which means that a CPU is connected to the DMP/VIP terminal. This message is the VIP mode prompt. It indicates that you are connected to the DMP and in VIP mode.

### TO DISPLAY CONFIGURATION PANELS

If you're having problems with a DPS 8 system and you suspect a bad switch setting, you can display the configuration panel of the unit you're concerned about.

1. The unit you want to work with must be connected to the DPU or the DMP/VIP. If you're using a DPU and you're not sure which unit is connected to it, follow the procedure described earlier in this section under "Finding Out Which Unit is Connected to the DPU". If the correct unit is connected to the DPU, proceed with step 2. If it isn't, follow the procedure described earlier in this section under "Changing the Unit Connected to the DPU". If you're using a DMP/VIP and the unit you want to work with isn't connected to it, follow the procedure described earlier in this section under "Getting Connected to the DMP/VIP".
2. If you're using a DPU, get into VIP mode.

3. If you want to display the configuration panel of a CPU or an SCU, press function key 1 (the key marked "F1") or type:

CF

If you want to display the configuration panel of an IOM, type:

CFG

4. Panel displays which exceed a full screen are automatically stopped at full screen intervals. To make a panel display continue, type a space. To make a panel display stop, press the DEL key. Do not try to interrupt a display before it finishes filling the screen.
5. If you're using a DPU and you want to display another unit's configuration panel, return to ready mode, then repeat steps 1-3. If you're using a DMP/VIP, just repeat steps 1-3.

### DPU/DMP COMMAND SUMMARY

The following summary of commands is not complete. It lists only those commands that are discussed in this manual. These commands should be sufficient for you to perform all the tasks you need to accomplish with the DPU and the DMP/VIP. If you need more information, you should contact CSD.

#### Ready Mode Commands (C? Prompt)

OFL - enters TM mode. You can also use this command to change the unit connected to the DPU.

? - gives you help with DPU commands.

#### TM Mode Commands (OFL? Prompt)

.U - shows you which unit is currently connected to the DPU.

VIP - enters VIP mode.

SUSP - exits TM mode, retaining the step condition. You must use this command if you are putting more than one processor in step at a time.

QUIT - exits TM mode and returns to ready mode.

<BRK> - exits TM mode and returns to ready mode.



## VIP Mode Commands (< UNIT > CMD Prompt)

### GENERAL COMMANDS

TM - exits VIP mode and returns to TM mode.

<BRK> - exits VIP mode and returns to TM mode.

### CPU COMMANDS

<F1> - displays the configuration switches of the CPU.

CF - displays the configuration switches of the CPU. \*

ST CU - places a processor in step.

BCE 24000 - causes an EXECUTE SWITCHES function to occur, and automatically takes the bootload processor out of step and restarts it.

BCE 24002 - forces an ESD

### SCU COMMANDS

<F1> - displays the configuration switches of the SCU.

CF - displays the configuration switches of the SCU.

### IOM COMMANDS

CFG - displays the configuration switches of the IOM.



## SECTION 35

# FAILURES THAT CRASH THE SYSTEM

System failures come in two main varieties: those which crash the system and those which don't. When a failure crashes the system, the system returns to BCE. When a failure doesn't crash the system, the system doesn't return to BCE. Failures which crash the system are described in this section. Failures which don't crash the system are described in Section 36. \*

### TO RECOGNIZE A SYSTEM CRASH IN AUTOMATIC MODE

When a failure crashes the system in automatic mode, the system performs automatic recovery procedures. These procedures do the following:

1. Dump Multics and the disks (using the "dump" command).
2. Perform an emergency shutdown (using the "esd" command). \*

You'll know the system is performing these procedures, because you'll see the following messages on the bootload console:

```
Dump # 1332
proc 1, dbr = 123456712345671234567123
.
.
.

begin emergency shutdown part 1
emergency shutdown part 1 complete
shutdown complete
```

### TO RESPOND TO A SYSTEM CRASH IN AUTOMATIC MODE

1. If automatic recovery procedures are successful and the system is in unattended mode, it may also reboot itself (using the "boot" command). If it does, you'll see the same messages you see when you boot Multics as part of bringing the system up. In this case, all you need to do is fill out an operations error report form.
2. If automatic recovery procedures are successful and the system is in attended mode (or in unattended mode and didn't reboot itself), you should reboot the system yourself. Type:

```
ec auto star
```

if your site uses the auto exec\_com, or:

```
boot star
```

if your site uses the boot command, where "star" is the short form of the startup command. The system will respond with the same messages it gives you when you boot Multics as part of bringing up the system. Remember to fill out an operations error report form.

\*

## TO RECOGNIZE A SYSTEM CRASH IN MANUAL MODE

When a failure crashes the system in manual mode, the system stops and returns to BCE; i.e., you receive a BCE ready message on the bootload console:

```
bce (crash) 1552.8
```

## TO RESPOND TO A SYSTEM CRASH IN MANUAL MODE

1. If you want the system to perform the automatic recovery procedures described earlier (dumping and ESD), type:

```
ec rtb
```

2. The system will respond with the messages described earlier, then stop.
3. You should reboot the system yourself, by typing:

```
ec auto star
```

if your site uses the auto exec\_com, or:

```
boot star
```

if your site uses the boot command, where "star" is the short form of the startup command.

4. The system will respond with the same messages it gives you when you boot Multics as part of bringing up the system. Remember to fill out an operations error report form.
5. If you want to recover the system yourself, see "To Recover the System" in Section 37.

## PROBLEMS

1. If any one of the automatic recovery procedures fails, you will have to finish it yourself, then continue with any remaining procedures. They will not restart after you've fixed the one that didn't work; they cease to run automatically once one of them fails. See Section 37, "Automatic Recovery Failures", for a description of how to recover the system yourself.

## SECTION 36

# FAILURES THAT DO NOT CRASH THE SYSTEM

There are basically two kinds of failures that don't crash the system. One is a result of the system *looping*. The other is a result of the initializer process *hanging*. In both cases, the system does not shutdown or return to BCE normally.

### TO RECOGNIZE A SYSTEM LOOP

The following events are all signs that the system is looping:

- the system stops responding to your commands
- all user terminals hang
- neither the system administrator nor any users can log in
- users who are logged in can't log out
- you receive a large number of messages which look something like this:

```
pxss: notify timeout, event = 144163153167,  
      processid = 367256147361
```

- on a L68 system, the lights on the control panels of all of the processors are steadily lit (i.e., they stop blinking)
- on the system indicator panel (if your site has one), all lights for all running CPUs are steadily lit except for the bottom one

### TO RECOGNIZE A SYSTEM HANG

The following events are all signs that the initializer process is hanging:

- the rest of the system seems to be fine, but the initializer doesn't respond to your commands (i.e., the bootload console doesn't respond, or it responds, but the commands you issue don't seem to get executed)
- users can't log in and out

### TO RETURN TO BCE

If you think the system is looping or the initializer process is hanging, the first thing you should do is consult your system administrator or system maintainer. He or she must determine that this is actually what's happening before you perform the procedures below.

When the system is looping or the initializer process is hanging, you must force the system to return to BCE. (In other words, you must crash the system.) There are two ways of doing this. One is known as *executing fault*, while the other is known as *executing switches*. You should always try executing fault first. If that doesn't work, try executing switches. Remember -- *never* attempt to execute fault or execute switches without consulting your system administrator or system maintainer first.

### Executing Fault

1. Go to the CPU which is in trouble. (If you can't figure out which CPU this is, go to the bootload processor. If you've deleted the original bootload CPU dynamically, be sure to use the CPU which the system now considers to be the bootload processor.) If it's a DPS 8, continue with step 2. If it's a Level 68, set the EXECUTE SWITCHES/EXECUTE FAULT switch on the display panel to EXECUTE FAULT. Then continue with step 2.

2. Press the EXECUTE button. On a Level 68, this button is located on the display panel. On a DPS 8, it's located on the configuration panel.

3. This action will send a fault into the system, which hopefully will "kill" the system.

4. If executing fault is successful, the system will respond by printing this message:

```
sys_trouble: execute fault by operator
```

and returning to BCE.

If the system is in automatic mode, continue with step 5. If the system is in manual mode, continue with step 7.

5. The system will ask you the following question:

```
Should normal recovery procedures be used?
```

6. If you want the system to perform automatic recovery procedures, type "yes." The system will do a dump and an ESD, then wait for you to reboot Multics. If you want to perform recovery procedures yourself, type "no," then see "To Recover the System" in Section 37. In either case, stop here. Do not continue with step 7.

7. The system will give you a BCE ready message on the bootload console:

```
bce (crash) 0836.2:
```

8. If you want the system to perform automatic recovery procedures, type:

```
ec rtb
```

The system will do a dump and an ESD, then wait for you to reboot Multics. If you want to perform recovery procedures yourself, see "To Recover the System" in Section 37.

## Executing Switches on a Level 68 System

### PLACING NONBOOTLOAD CPUS IN STEP

1. Place all of the CPUs *except* the bootload processor in MEM step. Do this by setting the CYCLE switch on the maintenance panel of each CPU except the bootload processor to MEM.

### EXECUTING SWITCHES

2. Go to the bootload processor. (If you've deleted the original bootload CPU dynamically, be sure to use the CPU which the system now considers to be the bootload processor.) Set the 36 DATA switches on the maintenance panel to the binary representation of "024000717200." To do this, set the following switches to 1: 4, 6, 18, 19, 20, 23, 24, 25, 26, 28. Set all other switches to 0.
3. Set the EXECUTE SWITCHES/EXECUTE FAULT switch on the display panel to EXECUTE SWITCHES.
4. Press the EXECUTE button on the display panel.
5. This action will tell the processor to execute the instructions located at the memory address you set in the DATA switches. \*

### RESULTS OF EXECUTING SWITCHES

6. If executing switches is not successful, the system will not return to BCE. If this happens, double-check the DATA switches to be sure they are set to "024000717200." If they aren't, start over with step 1.
7. If the system still doesn't return to BCE, follow the procedure described under "Executing Switches with a Different Bootload CPU (L68)" later in this section.
8. If executing switches is successful, the system will respond by returning to BCE. If the system is in automatic mode, continue with step 9. If the system is in manual mode, continue with step 11.
9. The system will ask you the following question:  

Should normal recovery procedures be used?
10. If you want the system to perform automatic recovery procedures, type "yes." The system will do a dump and an ESD, then wait for you to reboot Multics. If you want to perform recovery procedures yourself, type "no," then see "To Recover the System" in Section 37. In either case, continue with step 13.
11. The system will give you a BCE ready message on the bootload console:  

bce (crash) 0836.2:

12. If you want the system to perform automatic recovery procedures, type:

ec rtb

The system will do a dump and an ESD, then wait for you to reboot Multics. If you want to perform recovery procedures yourself, see "To Recover the System" in Section 37.

#### *TAKING NONBOOTLOAD CPUS OUT OF STEP*

13. If the system is doing automatic recovery procedures, wait until it has finished all dump and ESD attempts and is ready to reboot Multics. If you are recovering the system yourself, finish dumping Multics and performing emergency shutdown before you take the processors out of step.
14. Take all of the nonbootload CPUs out of step by setting the CYCLE switch on the maintenance panel of each one to OFF.

#### *INITIALIZING NONBOOTLOAD CPUS*

15. Initialize all of the nonbootload CPUs by pressing the INITIALIZE & CLEAR button on the configuration panel of each one at least 4 times. Each time you press the INITIALIZE & CLEAR button, wait about 5 seconds for the CPU to finish before you press it again.

#### **Executing Switches on a DPS 8 System**

#### *PLACING NONBOOTLOAD CPUS IN STEP*

1. Place all of the CPUs *except* the bootload processor in MEM step. Do this by turning off the PORT ENABLE switches on each CPU except the bootload processor.

#### *PLACING BOOTLOAD CPU IN STEP*

2. If you have a DPU, continue with step 3. If you have one DMP/VIP, continue with step 5. If you have multiple DMP/VIPs, continue with step 6.
3. The bootload processor must be connected to the DPU. (If you've deleted the original bootload CPU dynamically, be sure to use the CPU which the system now considers to be the bootload processor.) If you're not sure that the bootload processor is connected to the DPU, follow the procedure described in Section 34 under "Finding Out Which Unit is Connected to the DPU." If the bootload processor is connected to the DPU, proceed with step 4. If it isn't, follow the procedure described in Section 34 under "Changing the Unit Connected to the DPU." Then proceed with step 4.
4. Get into VIP mode. Continue with step 7.



5. Make sure that the DMP/VIP is connected to the bootload processor. See "Getting Connected to the DMP/VIP" in Section 34. (If you've deleted the original bootload CPU dynamically, be sure to use the CPU which the system now considers to be the bootload processor.) Continue with step 7.
6. Make sure you're using the DMP/VIP connected to the bootload processor. (If you've deleted the original bootload CPU dynamically, be sure to use the CPU which the system now considers to be the bootload processor.) Continue with step 7.
7. At the DPU or the DMP/VIP, put the bootload processor in CPU step by typing:

ST CU

#### *EXECUTING SWITCHES*

8. At the DPU or the DMP/VIP, type:

BCE 24000

This command will automatically take the bootload CPU out of step and restart it.

#### *RESULTS OF EXECUTING SWITCHES*

9. If executing switches is not successful, the system will not return to BCE. If this happens, follow the procedure described under "Executing Switches with a Different Bootload CPU (DPS 8)" later in this section.
10. If executing switches is successful, the system will respond by returning to BCE. If the system is in automatic mode, continue with step 11. If the system is in manual mode, continue with step 13.
11. The system will ask you the following question:

Should normal recovery procedures be used?
12. If you want the system to perform automatic recovery procedures, type "yes." The system will do a dump and an ESD, then wait for you to reboot Multics. If you want to perform recovery procedures yourself, type "no," then see "To Recover the System" in Section 37. In either case, continue with step 15.
13. The system will give you a BCE ready message on the bootload console:

bce (crash) 0836.2:

14. If you want the system to perform automatic recovery procedures, type:

ec rtb

The system will do a dump and an ESD, then wait for you to reboot Multics. If you want to perform recovery procedures yourself, see "To Recover the System" in Section 37.

#### *TAKING NONBOOTLOAD CPUS OUT OF STEP*

15. If the system is doing automatic recovery procedures, wait until it has finished all dump and ESD attempts, and is ready to reboot Multics. If you are recovering the system yourself, finish dumping Multics and performing emergency shutdown before you take the processors out of step.
16. Take all of the nonbootload CPUs out of step by turning on the PORT ENABLE switches on each one for all ports being used.

#### *INITIALIZING NONBOOTLOAD CPUS*

17. Initialize all of the nonbootload CPUs by pressing the INIT-CLEAR button on the configuration panel of each one at least 4 times. Each time you press the INIT-CLEAR button, wait for the CPU self tests to finish before you press it again. When the self tests are finished, the green light goes on. This light is located on the right hand side of the MP board (the topmost board in the CPU cabinet), just behind the free-edge connector for that board. Since the light is inside the CPU cabinet in an awkward position, it may be easier for you to simply wait 10 to 15 seconds before you press the INIT-CLEAR button again.

#### *Executing Switches with a Different Bootload CPU (L68)*

#### *PLACING CURRENT BOOTLOAD CPU IN STEP*

1. Put the current bootload processor (the one on which the first EXECUTE SWITCHES failed) in MEM step by setting the CYCLE switch on the maintenance panel to MEM.

#### *SETTING SWITCHES ON SCUS*

2. Pick a CPU to be the new bootload CPU. Then set switches on the configuration panel of each SCU as follows. Set the MODE switch to MANUAL. Set the MASK A MASK/PORT ASSIGNMENT switch to the number of the SCU port to which the new bootload CPU is connected. (*Note:* this is *NOT* the number of the new bootload CPU.) Set the PORT ENABLE switches which correspond to the new bootload CPU and all of the IOMs to ON.

### *TAKING NEW BOOTLOAD CPU OUT OF STEP*

3. Take the new bootload processor out of step by setting the CYCLE switch on the maintenance panel to OFF. Leave all other CPUs in step.

### *INITIALIZING NEW BOOTLOAD CPU*

4. Initialize the new bootload processor by pressing the INITIALIZE & CLEAR button on the configuration panel at least 4 times. Each time you press the INITIALIZE & CLEAR button, wait about 5 seconds for the CPU to finish before you press it again.

### *EXECUTING SWITCHES*

5. Go to the new bootload processor. Set the 36 DATA switches on the maintenance panel to the binary representation of "024000717200." To do this, set the following switches to 1: 4, 6, 18, 19, 20, 23, 24, 25, 26, 28. Set all other switches to 0.
6. Set the EXECUTE SWITCHES/EXECUTE FAULT switch on the display panel to EXECUTE SWITCHES.
7. Press the EXECUTE button on the display panel.
8. This action will tell the processor to execute the instructions located at the memory address you set in the DATA switches.

### *RESULTS OF EXECUTING SWITCHES*

9. If executing switches is not successful, the system will not return to BCE. If this happens, double-check the DATA switches to be sure they are set to "024000717200."
10. If the system still doesn't return to BCE, stop here and ask your system maintainer for help. Do not continue with this procedure.
11. If executing switches is successful, the system will respond by returning to BCE. If the system is in automatic mode, continue with step 12. If the system is in manual mode, continue with step 14.
12. The system will ask you the following question:  

Should normal recovery procedures be used?
13. If you want the system to perform automatic recovery procedures, type "yes." The system will do a dump and an ESD, then wait for you to reboot Multics. If you want to perform recovery procedures yourself, type "no," then see "To Recover the System" in Section 37. In either case, continue with step 16.
14. The system will give you a BCE ready message on the bootload console:

bce (crash) 0836.2:

15. If you want the system to perform automatic recovery procedures, type:

ec rtb

The system will do a dump and an ESD, then wait for you to reboot Multics. If you want to perform recovery procedures yourself, see "To Recover the System" in Section 37.

#### *TAKING NONBOOTLOAD CPUS OUT OF STEP*

16. If the system is doing automatic recovery procedures, wait until it has finished all dump and ESD attempts and is ready to reboot Multics. If you are recovering the system yourself, finish dumping Multics and performing emergency shutdown before you take the processors out of step.
17. Take all of the nonbootload CPUs out of step by setting the CYCLE switch on the maintenance panel of each one to OFF.

#### *INITIALIZING NONBOOTLOAD CPUS*

18. Initialize all of the nonbootload CPUs by pressing the INITIALIZE & CLEAR button on the configuration panel of each one at least 4 times. Each time you press the INITIALIZE & CLEAR button, wait about 5 seconds for the CPU to finish before you press it again.

#### *SHUTTING DOWN BCE*

19. Shut down BCE by typing:

die

#### *SETTING SWITCHES ON SCUS*

20. Set switches on the configuration panel of each SCU as follows. Set the MODE switch to PROGRAM. Set the MASK A MASK/PORT ASSIGNMENT switch to the number of the SCU port to which the bootload CPU is connected. (*Note:* this is *NOT* the number of the bootload CPU.) Set the PORT ENABLE switches which correspond to the bootload CPU and all of the IOMs to ON. Remember that the configuration will be different from what it was the last time you booted the system if it turned out that you had a bad bootload CPU.

#### *REBOOTING BCE*

21. Reboot BCE. See Section 11.

## Executing Switches with a Different Bootload CPU (DPS 8)

### *PLACING CURRENT BOOTLOAD CPU IN STEP*

1. Put the current bootload processor (the one on which the first EXECUTE SWITCHES failed) in MEM step by turning off its PORT ENABLE switches.

### *SETTING SWITCHES ON SCUS*

2. Pick a CPU to be the new bootload CPU. Then set switches on the configuration panel of each SCU as follows. Set the MODE switch to MANUAL. Set the MASK A MASK/PORT ASSIGNMENT switch to the number of the SCU port to which the new bootload CPU is connected. (*Note:* this is *NOT* the number of the new bootload CPU.) Set the PORT ENABLE switches which correspond to the new bootload CPU and all of the IOMs to ON.

### *TAKING NEW BOOTLOAD CPU OUT OF STEP*

3. Take the new bootload processor out of step by setting the PORT ENABLE switches to ON for all ports being used.

### *INITIALIZING NEW BOOTLOAD CPU*

4. Initialize the new bootload processor by pressing the INIT-CLEAR button on the configuration panel at least 4 times. Each time you press the INIT-CLEAR button, wait for the CPU self tests to finish before you press it again. When the self tests are finished, the green light goes on. This light is located on the right hand side of the MP board (the top-most board in the CPU cabinet), just behind the free-edge connector for that board. Since the light is inside the CPU cabinet in an awkward position, it may be easier for you to simply wait 10 to 15 seconds before you press the INIT-CLEAR button again.

### *PLACING NEW BOOTLOAD CPU IN STEP*

5. If you have a DPU, continue with step 6. If you have one DMP/VIP, continue with step 8. If you have multiple DMP/VIPs, continue with step 9.
6. The new bootload processor must be connected to the DPU. If you're not sure that it is, follow the procedure described in Section 34 under "Finding Out Which Unit is Connected to the DPU." If the bootload processor is connected to the DPU, proceed with step 7. If it isn't, follow the procedure described in Section 34 under "Changing the Unit Connected to the DPU." Then proceed with step 7.
7. Get into VIP mode. Continue with step 10.
8. Make sure that the DMP/VIP is connected to the new bootload processor. See "Getting Connected to the DMP/VIP" in Section 34. Continue with step 10.

9. Make sure you're using the DMP/VIP connected to the new bootload processor. Continue with step 10.
10. At the DPU or the DMP/VIP, put the new bootload processor in CPU step by typing:

ST CU

#### *EXECUTING SWITCHES*

11. At the DPU or the DMP/VIP, type:

BCE 24000

This command will automatically take the bootload CPU out of step and restart it.

#### *RESULTS OF EXECUTING SWITCHES*

12. If executing switches is not successful, the system will not return to BCE. If this happens, stop here and ask your system maintainer for help. Do not continue with this procedure.
13. If executing switches is successful, the system will respond by returning to BCE. If the system is in automatic mode, continue with step 14. If the system is in manual mode, continue with step 16.
14. The system will ask you the following question:  

Should normal recovery procedures be used?
15. If you want the system to perform automatic recovery procedures, type "yes." The system will do a dump and an ESD, then wait for you to reboot Multics. If you want to perform recovery procedures yourself, type "no," then see "To Recover the System" in Section 37. In either case, continue with step 18.
16. The system will give you a BCE ready message on the bootload console:  

bce (crash) 0836.2:
17. If you want the system to perform automatic recovery procedures, type:

ec rtb

The system will do a dump and an ESD, then wait for you to reboot Multics. If you want to perform recovery procedures yourself, see "To Recover the System" in Section 37.

### *TAKING NONBOOTLOAD CPUS OUT OF STEP*

18. If the system is doing automatic recovery procedures, wait until it has finished all dump and ESD attempts, and is ready to reboot Multics. If you are recovering the system yourself, finish dumping Multics and performing emergency shutdown before you take the processors out of step.
19. Take all of the nonbootload CPUs out of step by turning on the PORT ENABLE switches on each one for all ports being used.

### *INITIALIZING NONBOOTLOAD CPUS*

20. Initialize all of the nonbootload CPUs by pressing the INIT-CLEAR button on the configuration panel of each one at least 4 times. Each time you press the INIT-CLEAR button, wait for the CPU self tests to finish before you press it again. When the self tests are finished, the green light goes on. This light is located on the right hand side of the MP board (the topmost board in the CPU cabinet), just behind the free-edge connector for that board. Since the light is inside the CPU cabinet in an awkward position, it may be easier for you to simply wait 10 to 15 seconds before you press the INIT-CLEAR button again.

### *SHUTTING DOWN BCE*

21. Shut down BCE by typing:

die

### *SETTING SWITCHES ON SCUS*

22. Set switches on the configuration panel of each SCU as follows. Set the MODE switch to PROGRAM. Set the MASK A MASK/PORT ASSIGNMENT switch to the number of the SCU port to which the bootload CPU is connected. ( *Note*: this is *NOT* the number of the bootload CPU.) Set the PORT ENABLE switches which correspond to the bootload CPU and all of the IOMs to ON. Remember that the configuration will be different from what it was the last time you booted the system if it turned out that you had a bad bootload CPU.

### *REBOOTING BCE*

23. Reboot BCE. See Section 11.

### **Responding to the IOM Alarm**

If neither executing fault nor executing switches works, the problem may lie with the IOM. If the IOM alarm sounds, you have to reset it manually before you can return the system to BCE.

1. Turn off the alarm by pressing the ALARM RESET button on the front of the IOM.
2. On a Level 68 IOM, set the MAINTENANCE PANEL MODE switch on the configuration panel to TEST. Then press the (IOM INITIALIZE) MANUAL button on the maintenance panel. (Do *not* press the SYSTEM INITIALIZE button on the bootload panel.)
3. On a DPS 8 IOM, set the (IOM) TEST/NORMAL switch on the configuration panel to TEST. Then press the INITIALIZE button on the configuration panel. (Do *not* press the SYSTEM INIT button on the bootload panel.)
4. If your bootload console is a 6601, performing step 3 will put it into initialization. To correct this, press the CR button. The system will respond with:

CONSOLE READY...

5. Execute switches again.
6. Resetting the IOM may also reset a tape MPC. If this happens, you can take care of it by using the "fwload" command during system recovery. See "Reloading MPC Firmware" in Section 37.

### Forcing an ESD

If neither executing fault nor executing switches nor anything else you've tried has gotten the system to return to BCE, you can try forcing an ESD as a last resort. (Check with your system maintainer first.)

1. To force an ESD on a Level 68 system, follow the procedure described earlier under "Executing Switches on a Level 68 System." But in step 2, set the DATA switches to "024002717200" instead of "024000717200."
2. To force an ESD on a DPS 8 system, follow the procedure described earlier under "Executing Switches on a DPS 8 System." But in step 8, use the "BCE 24002" command instead of the "BCE 24000" command.

### PROBLEMS

1. If you can't get the system to return to BCE, the problem may lie with the low-order memory. If it does, you will have to reboot BCE. See Section 11.
2. If hardware problems are the root of a system failure, none of the procedures above will be successful. Hardware problems can be hard to detect. Sometimes, when a module is in trouble, the TROUBLE light on its control panel will light up. But this isn't always the case. If you suspect that one of the modules is having hardware problems, you should notify CSD. Remove the module from the system and reconfigure the system before rebooting it. Refer to Section 32, "Doing Dynamic Reconfiguration."



## SECTION 37

# AUTOMATIC RECOVERY FAILURES

When automatic recovery procedures fail, the system doesn't take a dump, perform ESD, and reboot itself. Instead, it waits for you to recover the system by doing these things yourself.

### TO RECOVER THE SYSTEM

#### Dumping Multics

1. Type:

```
ec dump
```

which will write a Multics dumps into the DUMP partition. A Multics dump contains the contents of the main system memory and the system tables.

2. The system will respond with:

```
Dump # 1332  
proc 1, dbr=123456712345671234567123  
  
bce (crash) 1552.5:
```

#### Performing ESD

3. Type:

```
esd
```

which will start an emergency shutdown (ESD) of Multics. Since an ESD destroys the current main memory image, you should only invoke the esd command *after* you've invoked the dump exec\_com (i.e., after you've taken a Multics dump). *DO NOT* attempt an ESD if any SCU has malfunctioned seriously or has been cleared to zero. When you do an ESD, memory gets written to disks. If an SCU has been cleared, the records on some disks will be zeroed. If you have any doubts, talk to your system maintainer. You also should not attempt an ESD if disk packs have been moved to a disk drive on another MPC.

4. The system will respond with:

```
begin emergency shutdown part 1
emergency shutdown part 1 complete
shutdown complete
```

```
bce (boot) 1554.9:
```

5. During the ESD, the system will return briefly to Multics. If the ESD is successful, the system will reenter BCE in one to four minutes, depending on the size of the system at your site.

### Reloading MPC Firmware

6. If a tape or unit record MPC was reset, type:

```
fwload <MPC name>
```

for example:

```
fwload mtpa
```

- \* 7. The system will respond with:

```
Booting mtpa on 10M a chn 22...
bce (boot) 1555.7
```

If a disk MPC needs to be reloaded, you will have to reboot BCE. Ask your system maintainer for help.

### Rebooting Multics

If your site has a procedure for cleaning up incremental dump tapes, you should perform that procedure before continuing with step 8.

8. Type:

```
ec auto star
```

if your site uses the auto exec\_com, or:

```
boot star
```

if your site uses the boot command, where "star" is the short form of the startup command. This will bootload Multics.

9. The system will respond with the same messages it gives you when you bootload Multics as part of bringing the system up. You should respond to these messages the same way you do then. See "To Boot Multics From BCE" in Section 13.

## Recording the Failure

10. Remember to fill out an operations error report form as soon as possible.

## PROBLEMS

1. If dumping Multics fails, you can try it again. If it still fails, proceed with emergency shutdown.
2. If emergency shutdown fails, you can try it again. If it still fails, ask your system maintainer for help.
3. If rebooting Multics fails, you can try it again. If it still fails, refer to "Problems" in Section 13, "Bootloading Multics."
4. If you get the message "No space on RPV", the system will probably crash again. This time, when you reboot Multics, type:

```
boot rlvs
```

The system will perform various kinds of salvaging, then pause in ring 1.  
Type:

```
star
```

The system will continue rebooting Multics.



## SECTION 38

# STORAGE SYSTEM FAILURES

### TO RESPOND TO AN MPC FAILURE

Each MPC has a series of lights on it called *firmware lights*. When something is wrong with the MPC, none of these lights blink at all. In addition, the TRBL light may go on.

If an MPC fails and the disk or tape drives connected to it are cross barred, another MPC will pick up its load and there won't be any problems. However, if an MPC fails and the disk or tape drives connected to it are not cross barred, all of the disk or tape drives it controls will be bad. In this case, the first thing you should try is reloading the firmware into the MPC. If reloading the firmware doesn't correct the problem, your next action will depend on the kind of messages the system sends you. If you get messages which indicate that the MPC can't be loaded, ask your system maintainer for help. If you get messages which indicate that there are still disk or tape errors, or if you don't get any messages at all, check with your system maintainer. Then try reinitializing the firmware in the MPC. *Note:* if you didn't attempt to reload the firmware, do not attempt to reinitialize the firmware.

### Reloading Firmware into an MPC

*Note:* if operators are not allowed to use the sac command at your site, you will not be able to perform this procedure.

1. At a regular user terminal, log in as a regular user by typing:

```
l <your Person_id>
```

for example:

```
l MacKenzie
```

where "l" is the short form of the login command.

2. The system will respond with:

```
Password:
```

3. Type your password. The system will respond with:

```
MacKenzie.Operator logged in 06/01/83 0921.4 est Wed
from VIP7801 terminal "none".
Last login 03/18/83 0726.2 est Fri from VIP7801
terminal "none".
r 09:22 3.0 3.50
```

where "r 09:22 3.0 3.50" is a ready message.

4. Type:

```
sac load_mpc <mpc name> -firm
```

for example:

```
sac load_mpc mspa -firm
```

where "mspa" is the name of the MPC as it appears in the config deck. *Note:* use of this command requires that your system administrator give your User\_id special privilege.

5. The system will respond with:

```
RCP: Attached dsk_a_00 for Initializer.SysDaemon.z
using >system_library_tandd>ifad_25.69
```

6. The system may then ask you a question:

```
load_mpc: Multiple revisions of disc 500 firmware for msp
model 609. Choose from g2, k1, l1, f2 or no.
```

7. Answer the question according to your system maintainer's instructions.

8. The system will load the firmware and you will get messages like these:

```
load_mpc: Loading 603000.dsc500.d500.k1 control store.
load_mpc: Loading 603000.dsc500.d500.k1 read/write.
RCP: Detached dsk_a_00 from Initializer.SysDaemon.z r
10:04 4:110 89
```

9. Log out as a regular user by typing:

```
logout
```

10. The system will respond with:

```
MacKenzie.Operator logged out 06/01/83 0928.3 est Wed
CPU usage 13 sec, memory usage 1.8 units, cost $0.28
hangup
```

11. If this procedure doesn't work, you may have to return to BCE. Ask your system maintainer for help.

## Reinitializing Firmware in an MPC

1. Press the INITIALIZE button on the MPC configuration panel.
2. Set the thumbwheel switches on the MPC configuration panel to "0484".
3. Press the RESET & BRANCH button on the MPC configuration panel.
4. If this procedure doesn't work, repeat steps 1 through 3 with the thumbwheel switches set to "0480".

## TO RESPOND TO DISK PROBLEMS

1. If there are problems with just one disk pack, you will notice a burst of error messages for the drive the pack is mounted on. Then the audible alarm will sound and you will receive a message like this:

```
disk_control: dska_04 requires intervention.
```

2. If a user is trying to access data on the drive, you will receive a message on the bootload console like the following:

```
pxss: notify timeout, event = 144163153167,  
processid = NNNNNNNNNNNN
```

If the processid is "003000777777", the initializer is waiting and no operator commands can be issued until the drive is made operational.

3. If the drive has dropped into standby, and the system hasn't crashed, you should attempt to ready the drive by pressing the READY button.
4. If this attempt is successful, and the drive becomes operational, the system will respond with:

```
disk_control: dska_04 now operational.
```

5. If this attempt is not successful, the system will continue to sound the audible alarm and send you messages. In this case, you should crash the system by executing fault, then perform an ESD. (See "Executing Fault" and "Performing ESD" in Sections 36 and 37 respectively.)

## TO MOVE 451 PACKS

If a 451 disk drive develops problems, you can try moving its packs and its address plugs to a different drive. Packs can only be swapped between two drives attached to the same MPC. This procedure can *only* be done with 451 drives. \*

**WARNING:** Don't use this procedure if you see disk fragments (dust particles) around the well of the original disk drive, or if you see other evidence of damage to the disk pack or original disk drive (e.g., a disk access arm head crash). Moving a damaged pack can cause further damage to the new disk drive. Consult your System Maintenance staff or a CSD representative if you suspect that the disk pack have been physically damaged.

1. Put the drive you are switching from (the old drive) in STANDBY mode by pushing the STOP button. If there is a disk volume on the drive you are switching to (the new drive), put that drive in STANDBY mode as well.
2. Remove the address plugs from both the old drive and the new drive immediately.
3. When the spindles stop, demount any volume from the new drive, and move the volume from the old drive to the new drive.
4. Push the START button to start the new drive.
5. After the new drive becomes ready, insert the address plug from the old drive into the new drive; insert the address plug from the new drive into the old drive.

Note that the new drive must already be ready before you insert the address plug. Note also that you must wait at least 30 seconds from the time you remove the plugs until the time you reinsert the plugs. This is because the MPC only polls devices every 15 seconds for status changes.

If the procedure above fails, the following procedure may be tried as an alternate:

1. Put the drive you are switching from (the old drive) into offline mode, by setting the rotary switch on the inside of the back door of the unit.
2. Power down the old drive.
3. Dismont the disk pack from the old drive and mount it on the drive you are switching to (the new drive).
4. Swap the address plugs between the two drives.
5. Put the new drive into offline mode, ready it, and then put it into online mode.



## SECTION 39

# FNP FAILURES

Usually, if an FNP crashes, the answering service reloads it automatically. But if this doesn't happen, or if there are problems with an FNP which don't cause it to crash, you should take it out of service. This involves sending users a warning, then dumping or shutting down the FNP, which disconnects the users logged in over its channels. When the dump or shutdown is complete, you must reload the FNP yourself.

### TO TAKE AN FNP OUT OF SERVICE

#### Warning Users

1. Type:

```
w * * <"message">
```

for example:

```
w * * "Taking FNP A out of service. You may be  
disconnected."
```

where "w" is the short form of the warn command.

2. The system will respond with:

```
Ready (MacKenzie)
```

#### Dumping or Shutting Down the FNP

3. If you've been asked to dump the FNP, type:

```
dump_mpx <FNP tag>
```

for example:

```
dump_mpx a
```

This will dump the FNP, and then shut it down automatically.

4. The system will respond with:

```
as_mcs_mpx_: Dump of FNP a created in  
>dumps>fnp.a.060183.1519
```

where ">dumps>fnp.a.060183.1519" is the pathname of the segment containing the dump.

5. If you've been asked to shut the FNP down, type:

```
shutdown_mpx <FNP tag>
```

for example:

```
shutdown_mpx a
```

6. The system will respond with:

```
Ready (MacKenzie)
```

7. If there are any users logged in over channels connectd to the FNP you're trying to dump or shut down, you'll get an error message. If you must do the dump or the shutdown, type:

```
dump_mpx a -force
```

or:

```
shutdown_mpx a -force
```

Otherwise, ask your system maintainer for help.

## TO RELOAD AN FNP

1. Type:

```
load_mpx <FNP tag> -force
```

for example:

```
load_mpx a -force
```

2. The system will respond with:

```
FNP A loaded successfully
```

3. Let users know that the FNP is back in service by typing:

w \* \* <message>

for example:

w \* \* FNP A is back up.

where "w" is the short form of the warn command.

4. The system will respond with:

Ready (MacKenzie)



## SECTION 40

# MANAGING THE DATA MANAGEMENT DAEMON

*Data Management* is a system which provides users with files that are protected against user process failures and Multics system failures. (They are not protected against hardware failures.) The *Data Management daemon* maintains the consistency and integrity of Data Management files by "undoing" unfinished modifications made to the files by user processes which have failed or which ran during a previous bootload session. The Data Management daemon also makes sure that the Data Management system's internal tables are correct.

### TO LOG IN THE DATA MANAGEMENT DAEMON DRIVER

1. Type:

```
login Data_Management.Daemon <DM driver label>
```

for example:

```
login Data_Management.Daemon dml
```

where "dml" is your site's label for the daemon which controls the Data Management system.

2. The system will respond with:

```
as LOGIN Data_Management.Daemon dmn dml (create)
dml Initializing Data Management in ring 2.
dml Data Management System initialized.
```

### TO SHUTDOWN THE DATA MANAGEMENT SYSTEM

1. Type:

```
r <DM driver label> shutdown
```

for example:

```
r dml shutdown
```

where "dml" is your sites label for the daemon which controls the Data Management system.

2. The system will respond with:

```
dm1 DM system shutdown scheduled to begin at 10/23/84
    1503.2, to bump users at 1513.2.
```

## PROBLEMS

1. If the Data Management system shutdown is unsuccessful, you may get a message like this:

```
dm1 dm_daemon (shutdown): Unable to schedule
    requested DM shutdown.
dm1 The current Multics shutdown time is earlier
    than the specified time.
```

or like this:

```
dm1 dm_daemon (shutdown): Unable to schedule
    requested DM shutdown.
dm1 No time was specified, and there is no Multics
    shutdown scheduled.
```

Inform the person who asked you to do the DM system shutdown that it was unsuccessful. In the case of the second message, tell the person that an explicit time is needed.

# APPENDIX A

## STARTUP CHECKLISTS OF SWITCH SETTINGS

This appendix contains checklists of switch settings for all of the major hardware modules described in this manual. You should check all of these switch settings before you bring the system up. It's very important that the switches be set correctly before the machines are run. It's especially important that the switch settings match the information in the configuration deck.

To make the task of checking switch settings easier, we suggest that you make copies of the checklists and tape them to the machines. Note any site differences.

## SCU CONFIGURATION PANEL SWITCHES

Switch	Position
PORT ENABLE	Ports for all IOMs and the bootload CPU ON (↑); other ports OFF (↓).
ALARM	ENABLE (↑)
LWR STORE SIZE	Set to size available for Store A and Store A1.
MODE	PROGRAM (↑)
MASK/PORT ASSIGNMENT	Mask A set to port for bootload CPU. Mask B set to OFF.
STORE A	ON LINE (↑) (if used).
STORE A1	ON LINE (↑) (if used).
STORE B	ON LINE (↑) (if used).
STORE B1	ON LINE (↑) (if used).
INTERLACE	ON (↑) if both stores equal in size; OFF (↓) otherwise.
LWR STORE	Whatever is desired.

All other switches on the SCU configuration panel should be in the OFF or down position.

## SCU MAINTENANCE PANEL SWITCHES

Switch	Position
MARGIN CONTROL MODE	PROGRAM (↑)

All other switches on the SCU maintenance panel should be in the OFF or down position. On DPS 8 systems, this panel has been replaced by a display.



## CPU CONFIGURATION PANEL SWITCHES

Switch	Position
PORT ENABLE	ON (↑) for ports used.
INITIALIZE ENABLE	ON (↑) for ports used.
ASSIGNMENT	000 (↓ ↓ ↓) for low-order memory; others as appropriate.
ADDRESS RANGE on L68; not present on DPS8	FULL (↑) or HALF (↓), as appropriate.
ALARM	NORMAL (↓) on L68; ENABLE (↑) on DPS8.
MAINTENANCE PANEL MODE	NORMAL (↓) on L68; PROCESSOR on DPS8.
FAULT CONTROL on L68; PROCESSOR FAULT BASE ADDRESS on DPS8	Switch 11 set to 1 (↑); all others set to 0 (↓) (100 octal).
PROCESSOR NUMBER	CPU A (PRO-0) = 000 (↓ ↓ ↓). CPU B (PRO-1) = 001 (↓ ↓ ↑). CPU C (PRO-2) = 010 (↓ ↑ ↓). CPU D (PRO-3) = 011 (↓ ↑ ↑). CPU E (PRO-4) = 100 (↑ ↓ ↓). CPU F (PRO-5) = 101 (↑ ↓ ↑). CPU G (PRO-6) = 110 (↑ ↑ ↓). CPU H (PRO-7) = 111 (↑ ↑ ↑).
OPERATING MODE	MULTICS on L68; VMS (↓) on DPS8.

All other switches on the CPU configuration panel should be in the OFF or down position.

## CPU MAINTENANCE PANEL SWITCHES

Switch	Position
ENABLE MATCH	Both switches ON (↑ ↑).
DATA SWITCHES	Switches 4, 6, 18, 19, 20, 23, 24, 25, 26, 28 set to 1 (↑); all others set to 0 (↓) (024000717200 decimal).
AUTO	OFF
CYCLE	OFF
EXECUTE PB/SCOPE REPEAT (Display panel)	EXECUTE PB (↑)
INITIALIZE & CLEAR/ INITIALIZE CONTROL (Display Panel)	INITIALIZE CONTROL (↓)
EXECUTE SWITCHES/ EXECUTE FAULT (Display Panel)	EXECUTE FAULT (↓)

All other switches on the CPU maintenance panel should be in the OFF or down position. On DPS8 systems, this panel has been replaced by a display.

## IOM CONFIGURATION PANEL SWITCHES

Switch	Position
PORT ENABLE	ON (↑) for ports used.
INITIALIZE ENABLE	ON (↑) for ports used.
ASSIGNMENT	000 (↓ ↓ ↓) for low-order memory; others as appropriate.
ADDRESS RANGE on L68; not present on DPS8	FULL (↑) or HALF (↓), as appropriate.
ALARM	NORMAL (↓) on L68; ENABLE (↓) on DPS8.
MAINTENANCE PANEL MODE on L68; IOM on DPS8	NORMAL (↓)
IOM BASE ADDRESS	IOM A = switches 14 and 15 set to 1 (↑); all others set to 0 (↓) (1400 octal). IOM B = switch 13 set to 1 (↑); all others set to 0 (↓) (2000 octal). IOM C = switches 13 and 15 set to 1 (↑); all others set to 0 (↓) (2400 octal). IOM D = switches 13 and 14 set to 1 (↑); all others set to 0 (↓) (3000 octal).
INTERRUPT BASE ADDRESS	Switches 14 and 16 set to 1 (↑); all others set to 0 (↓) (1200 octal).
IOM NUMBER	IOM A (IOM-0) = 00 (↓ ↓). IOM B (IOM-1) = 01 (↓ ↑). IOM C (IOM-2) = 10 (↑ ↓). IOM D (IOM-3) = 11 (↑ ↑).
SOURCE on L68; BOOT SOURCE on DPS8	TAPE (↓)

CHANNEL NUMBER CODE on L68;  
CHANNEL SELECT on DPS8

Channel number of tape controller which  
controls tape drive on which BCE/Multics  
tape is mounted.

ZERO BASE S.C. PORT NO. on  
L68; SCU PORT NUMBER on  
DPS8

SCU port to which IOM is cabled.

OPERATING MODE

PAGED

All other switches on the IOM configuration panel should be in the OFF or  
down position.

All switches on the IOM maintenance panel should also be in the OFF or  
down position. On DPS8 systems, this panel has been replaced by a display.

## FNP DIA SWITCHES (DN6670)

Switch	Position
6000 MAILBOX	
FNP A	↑ ↑ ↑ ↑ ↑ ↑    ↑ ↓ ↓ ↓ ↑ ↑ (3400 octal)
FNP B	↑ ↑ ↑ ↑ ↑ ↑    ↑ ↓ ↓ ↓ ↓ ↓ (3700 octal)
FNP C	↑ ↑ ↑ ↑ ↑ ↑    ↓ ↑ ↑ ↑ ↓ ↑ (4200 octal)
FNP D	↑ ↑ ↑ ↑ ↑ ↑    ↓ ↑ ↑ ↓ ↑ ↓ (4500 octal)
FNP E	↑ ↑ ↑ ↑ ↑ ↑    ↓ ↑ ↓ ↑ ↑ ↑ (5000 octal)
FNP F	↑ ↑ ↑ ↑ ↑ ↑    ↓ ↑ ↓ ↑ ↓ ↓ (5300 octal)
FNP G	↑ ↑ ↑ ↑ ↑ ↑    ↓ ↑ ↓ ↓ ↓ ↑ (5600 octal)
FNP H	↑ ↑ ↑ ↑ ↑ ↑    ↓ ↓ ↑ ↑ ↑ ↓ (6100 octal)
6000 TERMINATE	↑ ↓ ↓ ↑ (3 octal - ignore rightmost switch)
6000 EMERGENCY	↓ ↓ ↓ ↑ (7 octal - ignore rightmost switch)
HNP MAILBOX	↑ ↑ ↓ ↑ ↓ ↓ (454 octal)
HNP TERMINATE	↑ ↑ ↓ ↑ (2 octal)
HNP SPECIAL INTERRUPT	↑ ↑ ↓ ↓ (3 octal)

\*

Each FNP may be configured with one or two DIA boards. Each board must be configured on a separate FNP port. The FNP ports on which DIA boards may be configured are 3, 4, 5, and 14. Each board's FNP port must be cabled to an IOM channel. The FNP DIA connections are identified in the Multics config deck by the IOM channel to which the FNP port is cabled. The Multics software determines which FNP port to use in accessing the FNP by references to the IOM channel cabled to the active FNP port. The FNP port number is not recorded in the FNP core image, nor in Multics supervisor databases, nor in any Multics config card.

A FNP with two DIA boards can be cabled to two different IOMs on a single Multics system, or to an IOM on each of two different systems. However, only one of the DIA boards may be used at a time. The prph fnp config card for the IOM channel cabled to the active DIA must have a state of on; the card for the IOM channel cabled to the inactive DIA must have a state of off.

Cabling a FNP to two different IOMs on a single Multics system offers a measure of improved reliability. If the IOM attached to the active DIA board breaks down, the IOM and its attached FNP can be deleted from the system, and the FNP

can then be added to the system using the other IOM channel. However, users of the FNP at the time of the IOM failure will have to login again. If their processes had the save\_on\_disconnect attribute, they will be able to reconnect to their processes and continue the work which was interrupted when the IOM failed. If their processes did not have the save\_on\_disconnect attribute, work in progress when the IOM failed will be lost.

Cabling a FNP to the IOMs of two different systems allows the FNP to be shifted easily from one system to the other.

Multics requires that each FNP use a paging mechanism to access FNP memory beyond the first 32K words of memory. The paging mechanism on the FNP pager board can be disabled for testing purposes, but Multics requires that it be enabled during normal operations. Contact your CSD representative if your FNP will not operate. Ask him to insure that the paging mechanism is fully operative.

# APPENDIX B

## GLOSSARY

- absentee facility**  
the facility which controls absentee jobs.
- absentee process**  
a process which belongs to a person who is not logged in and is not interacting with the system. The person has prepared an entire job beforehand and is having it run at a specified time or when specified resources are available.
- active**  
the state of a major hardware module or peripheral when it is turned on in the config deck.
- admin.ec**  
an `exec_com` which is invoked any time an `exec ("x")` command is issued.
- administrative ring**  
ring 1.
- answering service**  
a subsystem which runs in the initializer process and performs these functions: handling dialups, logins and logouts, supervising system accounting, controlling interactive and absentee users, loading and dumping FNP's.
- answering service log**  
a log of messages produced by the answering service, containing information about all user and daemon logins and logouts.
- answering service process**  
the initializer process.
- argument**  
an additional piece of information typed after a command, on the same line.
- assigning a drive**  
the process of converting a disk drive from one kind of usage to another.
- attended mode**  
a mode in which the system assumes an operator is present to mount tapes and disks.
- attended service**  
attended mode.
- audible alarm**  
an alarm contained in the bootload console or the initializer terminal.

automatic mode

a mode in which the system automatically takes a dump and performs an ESD after a crash.

automatic reboot mode  
automatic mode.

auto exec\_com

an exec\_com which does what the boot command does, and also puts the system in automatic mode.

backup

those systems which ensure that user segments and directories can be recovered if they are destroyed due to system failure or user error.

backup daemon

a daemon which controls some part of the Multics backup system.

backup daemon command

one of the commands which control backup daemons.

backup dump

an incremental dump.

backup tape

an incremental tape.

banner bars

several lines of characters printed at the bottom of a headsheet.

BCE command

a command used to communicate with BCE.

BCE message

a message produced by BCE.

BCE/Multics system tape

the tape from which the programs needed to boot Multics are read.

BCE partition

a special area on the RPV in which BCE resides after it is booted.

beeper

the audible alarm.

booting

the process of loading a set of programs into memory, linking them together so they can refer to each other, setting up any necessary data bases, and running the programs to start up a system.

Bootload Command Environment (BCE)

a very simple operating system which provides a boot and crash environment for Multics, and performs the following functions: booting Multics, dumping main memory and disks, initiating emergency shutdown of Multics.



bootload console  
a special terminal that is connected to the bootload (low order) IOM or IMU. When more than one console is present, only the active one is called the bootload console.

bootload CPU  
the CPU that is selected by the bootload SCU.

bootloading  
bringing up the system.

bootload IOM  
the IOM that is selected by the bootload SCU.

bootload SCU  
the SCU that is configured with a base address of zero.

bootload session  
the period of time between a Multics bootload and a Multics shutdown.

box  
any major hardware module.

bringing the system up  
performing system startup.

calendar clock  
a clock contained in an SCU. The one in the bootload SCU is the one used by the system. It keeps accurate time, precise to the microsecond.

card punch  
an output device that provides punched card output in response to software programs.

card reader  
a device that transfers programs and data punched on computer cards to the central system.

catchup dump  
the operation of dumping to tape all system segments (files) that have been modified since the previous complete or catchup dump was performed. This is also known as a consolidated dump operation. See "Backup" in Section 3.

catchup tape  
one of the tapes created during a catchup dump operation.

central processing unit (CPU)  
a major hardware module responsible for performing most of the computational processing done by the system (i.e., performing most of the arithmetic and logical manipulations of data).

clock card  
a configuration card that provides information about how to interpret the readings of the calendar clock, and must be edited when the clock is changed for daylight savings time.

**cold boot**

a bootload that completely recreates the operating environment and/or generates a new storage system hierarchy.

**cold bootload**

a cold boot.

**combined card unit (CCU)**

a device that functions as both a card reader and a card punch.

**command**

an operator "command" is an instruction typed on the bootload console or on an initializer terminal that tells the system what to do or provides it with information.

**communications channel**

a communications path between a user terminal and the system. This path leads through the Front-End Network Processor (FNP). It is often called an FNP channel.

**communications line**

anything from a simple pair of wires in a circuit to the entire telephone system that is used by the FNP to provide a physical connection between the system and a terminal, a network, or another computer.

**communications processor**

a Front-End Network Processor (FNP), which routes information from many terminals through a single IOM channel into the system.

**complete dump**

a backup dump that locates and dumps every segment and directory in the storage system without regard for when they were last modified.

**complete dump tape**

a tape produced by a complete dump.

**config deck**

the configuration deck.

**configuration**

a three-part process which involves choosing hardware modules and connecting them with cables, setting switches on the hardware modules to connect them into an arrangement that can run Multics, and telling the software how to use the hardware.

**configuration deck**

data fed to the system from a tape or from the bootload console, which tells the software about the actual hardware configuration available, the switch settings of the major modules, the cabling between the modules, and the operational readiness of the hardware, the peripherals, and some of the software data bases.

**console**

the bootload console.

**consolidated dump**

a backup dump which locates and dumps all segments and directories which have been modified after some specified time in the past.

**consolidated tape**

a tape produced by a consolidated dump.

**control file**

a segment containing a list of the pathnames of segments and directories to be retrieved.

**coordinator**

the coordinator process.

**coordinator process**

the I/O daemon which controls driver processes.

**coordinator request number**

one of the numbers which identify a print request.

**cpu card**

a configuration card which identifies a processor and must be edited to change the state of a processor.

**crash**

a shutdown which is the result of a system malfunction, is unplanned, and causes the system to become unavailable to users.

**cross barred**

connected to more than one IOM (in the case of an MPC) or connected to more than one MPC (in the case of a disk or tape drive).

**cross retrieval**

a retrieval in which a segment is retrieved with a different pathname.

**crt terminal**

a video terminal.

CSU6001/6004 (6001/6004)

a bootload console which prints its input and output on paper.

CSU6601 (6601)

a bootload console which displays its input and output on a screen.

Customer Services Division (CSD)

comprises Honeywell employees who maintain the hardware and software. This division used to be known as Field Engineering.

daemon

a system service process.

daemon command

a command used to communicate with a daemon.

data base

a segment which contains data used by the system.

Data Management

a system which provides users with files that are protected against user process failures and Multics system failures.

Data Management daemon

a daemon which maintains the consistency and integrity of Data Management files by "undoing" unfinished modifications made to the files by user processes which have failed or which ran during a previous bootload session.

datanet

a Front-End Network Processor.

device class

a subdivision of a request type by access class, used for AIM.

density

a number which expresses the amount of information stored on a tape.

Diagnostics Processor Unit (DPU)

a stand alone computer system which serves as an interface to the DMP for the processor, the SCU and the IOM.

direct interface adapter (DIA)

the cable which provides the physical connection between an FNP and an IOM.

directory

a "catalogue" of segments and subdirectories.

directory hierarchy

the storage system hierarchy.

directory salvager

a salvager which can be run along with everything else in the system.

disk

the principal means of storing information on Multics.

disk controller  
an MPC that controls disk drives.

disk drive  
a mechanical device on which a disk pack is mounted for reading by the system.

disk pack  
an individual rotating storage unit upon which system files can be magnetically recorded. The disk pack can be removable or nonremovable. Nonremovable packs are often called fixed head disks or head assemblies.

distributed processing system (DPS) 8  
the new, improved Multics system.

dprint  
a daemon printout of a user's file.

driver  
an IO process that controls one or more input/output devices.

driver process  
an I/O daemon that controls a local or remote device.

dumping  
recording system memory or system files onto tape or disk. System memory is dumped into a file after a system crash (failure) to learn why the system failed. Storage system files are dumped to tape routinely to provide backup copies of data. These backup copies can be reloaded as a group if an entire disk drive fails, or retrieved individually if a user accidentally deletes the wrong file.

dynamic maintenance panel (DMP)  
the part of the DPS 8 processor that produces terminal displays.

dynamic reconfiguration  
the process of adding or deleting a major hardware module or peripheral to or from the Multics configuration when the system is at Multics level (i.e., while the system is running).

emergency shutdown (ESD)  
a program that runs after a crash, as part of the crash recovery process, that makes the contents of the storage system consistent.

erase character (#)  
a keystroke that erases one character, the character immediately preceding it.

exec\_com  
a segment containing a series of either Multics or BCE commands to be executed.

exec command  
a command that is included in the admin.ec.

**executing fault**

the first way of forcing the system to return to BCE: an interrupt is sent into the system.

**executing switches**

the second way of forcing the system to return to BCE: the bootload processor is told to execute the instructions located at the memory address set in the DATA switches.

**firmware**

specialized software that is attached to hardware but subordinate to it; a set of programs that replace some of the electronic circuits in a machine.

**firmware lights**

a series of lights on an MPC.

**FNP crash**

an unplanned failure in the operation of an FNP that causes terminals channeled through that FNP to be disconnected from the system. The user process connected to the terminal may still remain active if the process has the `save_on_disconnect` attribute.

**front-end**

a Front-End Network Processor.

**Front-End Network Processor (FNP)**

a major hardware module responsible for data communications (i.e., providing the logical and physical connections between the system and terminals, networks, and other computers).

**hardcopy terminal**

a terminal that prints input and output on paper.

**hardcore**

System supervisory software (the hardcore supervisor).

**hardcore message**

a message produced by system software, often reporting a hardware problem. It can be printed on the bootload console or on an initializer terminal; it can also be recorded permanently in a system log (e.g., the `syserr` log).

**hardcore partition**

a special area on disk in which the supervisor resides when the system is running.

**hardcore supervisor**

the part of the system software that performs the supervisory functions of the system. It can't be changed while the system is running, and includes programs that must be present to bring the system up and programs that run the storage system.

**hardware**

all of the physical devices and electronic circuitry in a computer system.

hardwired  
directly connected.

head-of-string device  
the device containing some controller logic for FIPS-style devices. These devices are connected to a controller in serial strings, and this device is the first in that string.

headsheets  
the beginning identification page of a dprint.

help file  
an online segment that contains a description of a command, procedure or concept.

hierarchy backup system  
a backup system that dumps segments and directories by following the storage system hierarchy. \*

high-speed printer  
a line printer connected directly to the IOM or IMU.

inactive  
the state of a major hardware module or peripheral when it is known to the system but is unused (turned off in the config deck).

incremental dump  
a backup dump that locates and dumps all segments and directories that have been created or modified since the last time an incremental dump was done.

incremental tape  
a tape produced by an incremental dump.

information multiplexer unit (IMU)  
a model of IOM that is controlled by a microprocessor instead of being hardwired.

initialization  
system startup.

initializer  
the initializer process.

initializer command  
a command used to communicate with the initializer process.

initializer command response  
a response to an initializer command.

initializer hang  
one of two kinds of system failures that don't crash the system.

initializer process

the system control process for Multics, created when Multics is booted and active as long as Multics is running. It's the only process that can create and destroy other processes. It performs these functions: bringing the system up, operating the answering service, servicing operator commands, handling user requests, managing system terminals and all other terminal channels, routing messages, taking care of administrative tasks, shutting the system down.

initializer terminal

\* a regular terminal that can be used to issue all operator commands except BCE commands.

input mode

a mode in which the bootload console is accepting input.

input/output multiplexer (IOM)

a major hardware module responsible for managing all of the peripherals connected to the system (i.e., handling all transfer of data between peripherals and memory).

integrated peripheral controller (IPC)

an IMU channel. IPCs connect IMUs to other controllers or to peripherals, such as consoles and FNPs.

interactive process

a process that belongs to a person using a terminal, who is exchanging information directly and immediately with the system.

I/O daemon

a special user process that has the attribute "daemon" and can be used to control printers, card readers, etc.

I/O daemon command

one of the commands that control I/O daemons.

IO Daemon logs

logs of messages produced by the I/O daemon driver processes, containing information about all I/O requests.

iom card

a configuration card that identifies an I/O mainframe and must be edited to change the state of an IOM or an IMU.

IOM channel

a connection between the IOM and an FNP, an MPC, or a console, over which the system can do I/O.

keyname

a name assigned to a major hardware module cabled to a DPU that uniquely identifies the module to the DPU.

keypunch machine

a typewriter-like device used to type characters onto and punch holes into computer cards.



kill character (@)  
a keystroke that erases a whole line, everything from the cursor back to the left-hand side of the screen.

killing a mount request  
denying the mounting of a tape or disk.

label  
a tape or disk identifier.

Level 68  
the model number of the Multics processor manufactured prior to the current 8/70M model.

line printer  
a high-speed printer that prints a line at a time (vs a character at a time). Normally connected directly to the IOM or IMU.

link adapter  
an MPC's connection to an IOM.

load control group  
a group of users created by a system administrator and controlled by limits he or she sets up.

lock mode  
a mode in which the bootload console is not accepting input.

logging  
a means of keeping track of system events. The system writes messages to system logs to record system status; these logs provide a history of activity on the system. Operators also keep hand-written logs recording significant system events (e.g., CPU or memory configuration changes, hierarchy backup tape numbers, events leading up to a system crash, new or changed operation procedures, etc.).

logical device  
used by a remote driver to control a physical device.

logical volume  
a set of physical volumes (disk packs) that have some logical relationship to each other.

LOG partition  
a reserved area of disk on which syserr messages are written.

low-order memory  
the bootload SCU.

magnetic tape processor (MTP)  
an MPC that controls tape drives.

**maintenance channel adapter (MCA)**

a micro processor in the IMU that controls the IMU, the system init and boot functions, and all firmware loading of micro processors in the IMU central and channels. It can also perform test and diagnostics on the IMU.

**major device**

a physical device as defined in the I/O daemon tables.

**manual mode**

a mode in which the system pauses in BCE after a crash and waits for you to take action.

**mass storage processor (MSP)**

an MPC that controls disk drives.

**master console (for multidrop interface (MDI) to IMUs)**

the console that handles the operator communications to and from the maintenance channel adapter(s) for the IMUs configured on the MDI. There can be only one master console per MDI, and only one MDI per system; this is designated by the internal configuration file in the IMUs.

**master directory**

a directory whose segments reside on a different logical volume than the segments of the directory immediately above it.

**match id**

a number associated with a print request.

**mem card**

a configuration card that identifies a system controller and must be edited to change the state of a system controller.

**memory**

the part of a Multics configuration that contains instructions and manipulated data.

**message coordinator**

a facility that distributes messages between daemons (I/O driver processes and other system daemon processes) and the bootload console and initializer terminals. It allows many daemon processes to be controlled from a limited number of terminals (virtual consoles). With the message coordinator, you don't need 25 terminals to run 25 daemon processes.

**message coordinator message**

a message produced by system control, a daemon, or the answering service.

**message coordinator name**

same as message coordinator source id.

**message coordinator source id**

a label printed in messages to identify which daemon process outputs the message. You can use the same id with the "reply" command to send input to that daemon. Examples: bk = Backup, vinc = Volume\_Dumper, prta = IO.SysDaemon (printer A).

message coordinator terminal  
a bootload console and any other configured initializer terminal.

microprogram  
firmware.

microprogrammed peripheral controller (MPC)  
a major hardware module that controls storage devices or unit record devices.

minor device  
a logical device as defined by the I/O daemon tables.

mounted  
the state of a physical volume when the pack(s) are housed in the drive, the drive is operationally ready, and this has been made known to Multics.

Multics (Multiplexed Information and Computing Service)  
a complex operating system.

Multics command  
a standard command used to communicate with Multics.

multiplexer  
a device for funneling many communications channels into a single, higher speed connection to the system. An FNP is a multiplexer: many terminals connect through the FNP to a single IOM channel. Certain channels within an FNP can also be multiplexers, connecting many terminals through a special communications device (e.g., a modem) to a single FNP channel.

network  
a collection of hardware that provides the service of connecting many different pieces of data processing equipment, allowing them to communicate with each other.

online software  
the part of the system software that can be changed while the system is running, and includes all of the programs that live in the storage system and use it.

operating system  
a set of programs that reside in a computer and do the following: interpret users' instructions, control the hardware, supervise the basic operation of the system, share system resources among many users.

operator console  
the bootload console connected to the system through a channel on either an IOM or an IMU.

orderly shutdown  
a shutdown that is normal and planned.

This page intentionally left blank.

**panel**  
a collection of switches and lights on a hardware device, used to control the configuration of the hardware and to make changes in the way the hardware operates.

**pathname**  
a name which specifies a segment's or directory's location in the storage system hierarchy.

**peripheral**  
a device connected to a Multics system configuration and controlled by it; a terminal, storage device, unit record device, FNP or bootload console.

**physical volume**  
a set of data accessed as a group.

**printer**  
an output device which produces hardcopy (paper) output in response to software programs.

**printing terminal**  
a hardcopy terminal.

**private logical volume**  
a storage system logical volume to which access is restricted and whose physical volumes are not permanently mounted.

**process**  
a user's share of the system, which begins when he logs in and ends when he logs out.

**process directory**  
a directory which contains segments that are only meaningful during the life of a process.

**processor**  
a Central Processing Unit.

**program**  
a set of instructions for solving a problem, coded in a language the computer understands.

**prph card**  
a configuration card which identifies a peripheral and must be edited to change the state of a bootload console.

**public logical volume**  
a storage system logical volume to which access is unrestricted and whose physical volumes are usually permanently mounted.

**ready mode**  
the first mode a DPU enters after it is booted, in which the unit connected to it can be changed.

recovery

the procedure by which the backup systems recover segments and directories that have been dumped onto tape and place them back into the storage system hierarchy.

reloading

the global recovery of a major portion of the hierarchy when it has been damaged.

remote device

a unit record device which is connected to the FNP and communicates with the IOM via the FNP.

remote driver

a remote driver process.

remote driver process

an I/O daemon which controls a remote device.

remote job entry (RJE)

a process whereby standard Multics commands are punched on cards, then copied into an absentee segment which is submitted as an absentee job.

remote station

a remote device which is physically remote from the Multics system.

request type

a set of queues into which users may place requests for the I/O daemon to perform services.

Resource Control Package (RCP)

a facility which controls and allocates peripheral resources by managing the assignment of all storage and unit record devices and any disk or tape volumes which can be mounted on them, producing mount messages, and checking users' access to to use devices and volumes.

retrieving

the recovery of individual segments and directories at the request of users.

ring 1

the first ring the initializer process enters when the system is started up.

ring 4

the second ring the initializer process enters when the system is started up.

rings

the eight levels of privilege in a Multics system. Lower numbered rings are more privileged than higher numbered rings.

RJE station

a remote station.

root

the root logical volume (RLV).

root logical volume (RLV)

the special logical volume (one or more physical volumes) that contains all the files needed to bring Multics up all the way to ring 4 command level, plus all the directories in the hierarchy. The information contained in the RLV must be available to the system.

root physical volume (RPV)

the single physical volume required to boot BCE, to bring Multics up as far as ring 1 command level, and to do ring 1 functions like reloading volumes and performing some kinds of crash recovery. It is a physical volume of the RLV.

salvagers

a set of programs that do cleanup work on the storage system, detecting damage and, if possible, correcting it.

segment

the basic unit of information stored in the storage system. All files and directories are stored inside segments. A segment can contain from 0 to 255 records of data; each record is 1024 machine words long, and can hold 4096 characters of data.

sentinel

a message from a daemon that begins with "-->" and indicates that the daemon wants input.

service session

a bootload session.

session

a bootload session.

software

all of the programs that control the activities of a computer system.

source id

same as message coordinator source id.

storage device

a disk or a tape.

storage director

the controller for FIPS disk devices. The head-of-string devices connect to it on the device side. The host side connects to the FIPS channel in the IMU. The firmware for these controllers cannot be loaded from the system.

storage system

the combination of hardware and software that Multics uses for storing information.

storage system disk

a disk that belongs to the system and contains some part of the storage system hierarchy in a standard format.

storage system hierarchy

the structure into which all segments and directories are organized. It looks like an upside-down tree, with a single directory called "the root" at the top and the rest of the directories spreading out like branches from there downward.

store unit

one of the parts into which the memory associated with an SCU is divided.

supervisor

the hardcore supervisor.

swapping ttys

rerouting messages from one terminal to another.

syserr log

a log of messages produced by the supervisor and some of the online programs.

syserr message

a message produced by a subsystem or program and contained in the syserr log.

system administrator

a person who is responsible for: providing his or her site with a particular Multics operating environment, controlling and allocating resources (including those managed by RCP), registering projects and users, creating load control groups, setting prices on resources, setting limits on and billing for resource usage, scheduling system activities such as hours of operation, shift change times, and unattended service, describing site parameters and setting site options, assuring system security. System administrators are responsible to upper management for the successful overall operation of the entire computer system. Thus, they set policies from which operators and system maintainers take their direction.

system console

the bootload console connected to the system through a channel on either an IOM or an IMU.

system control

the initializer process.

system controller

a System Controller Unit.

System Controller Unit (SCU)

a major hardware module responsible for: controlling and coordinating the activities of all other hardware modules, controlling memory, interfacing between memory and CPUs and IOMs.



system crash

an unplanned termination of system availability caused by problems with either the hardware or the software.

system log book

a physical "pencil and paper" log that is the operator's record of unusual occurrences.

system loop

one of two kinds of system failures that don't crash the system.

This page intentionally left blank.

**system maintainer**  
a system programmer who is responsible for: configuring and tuning the operating system to make it comply with special site requirements, backing up and recovering the system, salvaging and scavenging, analyzing crashes, balancing disks, setting up things like I/O daemons and the message coordinator, metering and tuning, maintaining system data bases.

**system shutdown**  
the process of ending Multics service.

**system software**  
the programs that come with Multics.

**system startup**  
the process of beginning Multics service.

**system\_start\_up.ec**  
an exec\_com which is invoked automatically when the answering service is started up. It initializes the answering service, and may also contain commands which turn on the message coordinator, log in the daemons, start up the absentee facility, load the FNPs, set the message of the day, schedule system shutdown, accept additional channels for the initializer process, and set up the disposition of various system messages.

**tag**  
a number or letter which identifies a particular box.

**tailsheet**  
the ending identification page of a dprint.

**tape**  
a means of storing information on Multics.

**tape controller**  
an MPC which controls tape drives.

**tape drive**  
a device which houses tape reels.

**tape handler**  
a tape drive.

**tape reel**  
an individual tape unit.

**terminal**  
a device used to send input to and receive output from the system, with a keyboard like that on a typewriter.

**test and diagnostics (T & D)**  
a collection of online programs used by CSD. The programs allow CSD to run tests which help them maintain maximum system availability.

transparent (TM) mode  
the second mode a DPU enters, in which the unit connected to it can be displayed.

unattended mode  
a mode in which the system assumes there is no one present to mount tapes or disks. User tape mount requests are not honored and user print requests line up in the queues.

unattended service  
unattended mode.

unit  
any major hardware module.

unit record controller  
an MPC which controls printers, card punches and card readers.

unit record device  
a printer, a card punch or a card reader.

Unit Record Processor (URP)  
an MPC which controls printers, card punches and card readers.

unlock mode  
input mode.

user control  
the answering service.

user I/O disk  
a disk which belongs to a user and can contain any kind of data in any format.

user ring  
ring 4.

User\_id  
a name assigned to a registered Multics user or a daemon which identifies him or it to the system.

video terminal  
a terminal which prints input and output on a television-like screen.

VIP mode  
the third mode a DPU enters, in which switches can be executed to return to BCE and processors can be put in and out of step. It is also the one mode in which the DMP/VIP operates.

volume backup system  
a backup system which dumps segments and directories by following the layout of files on disk packs, dumping segments and directories in an order unrelated to the storage system hierarchy.

volume salvager  
a salvager which must be the only thing running in the system.

volume scavenger  
a salvager which can be run along with everything else in the system.

wakeup dump  
an incremental dump.

wakeup tape  
an incremental tape.

warm boot  
a bootload which assumes that some information from the previous bootload is to be used and/or maintains the current storage system hierarchy.

warm bootload  
a warm boot.

workstation  
a computer which acts as a satellite for Multics, to which physically remote devices (remote stations) are connected.

x command  
an exec command.



# INDEX

- .U command (DPU) 34-3, 34-6
- 451 disk drive 2-5.1, 3-3, 8-1, 18-1, 20-1, 21-2, 21-3  
moving 38-3
- 500 disk drive 2-5.1, 3-3, 8-1, 18-1, 20-1, 21-2, 21-3
- 500 tape drive 8-1, 11-1, 11-2, 18-1, 19-3
- 501 disk drive 2-5.1, 3-3, 8-1, 18-1, 20-1, 21-2, 21-3
- 600 tape drive 11-1, 11-2, 19-3
- 610 tape drive 8-1, 11-1, 11-2, 18-1, 19-3
- 630 tape drive 11-1, 11-2, 19-3
- 3380 disk drive 2-5.1, 3-3, 8-1, 18-1
- 3381 disk drive 2-5.1, 3-3, 8-1, 18-1
- 6000 EMERGENCY switch (FNP) A-7
- 6000 MAILBOX switch (FNP) A-7
- 6000 TERMINATE switch (FNP) A-7
- 8200 tape drive 8-1, 11-1, 18-1, 19-4
- <BRK> key (DPU) 34-3, 34-6
- <F1> key (DMP/VIP) 34-6, 34-7
- <F1> key (DPU) 34-6, 34-7
- ? command (DPU) 34-4, 34-6
- A
- abs command (initializer)
  - bump 30-4
  - defer 30-5
  - list 30-7
  - maxq 30-4
  - maxu 30-2
  - move 30-3
  - release 30-6
- abs command (initializer) (cont.)
  - run 30-6
  - start 30-1
  - start queue 30-3
  - stop 17-4, 30-1
  - stop queue 30-3
  - suspend 30-5
- absentee
  - facility 3-10, 30-1, B-1
    - starting 30-1
    - stopping 17-1, 17-3, 30-1
  - jobs
    - bumping 30-4
    - deferring 30-5
    - forcing log in 30-6
    - getting info 30-7
    - moving 30-3
    - releasing 30-6
    - suspending 30-5
  - process 3-10, 3-16, B-1
  - queues
    - restarting 30-3
    - setting highest numbered 30-4
    - stopping 30-2
  - slots
    - setting 30-2
- accept command (initializer) 5-6, 6-3
- active state 12-1, B-1
- add\_lv command (initializer) 21-1, 21-3
- add\_vol command (initializer) 21-2, 21-4
- adding a CPU 29-1, 32-1
- adding a disk drive 20-5, 21-6
- adding a FNP 32-5
- adding a logical channel 32-6
- adding a SCU 29-1, 32-2
- adding a tape drive 19-5, 19-6
- adding alternate bootload console 32-7
- adding an IMU 32-4
- adding an IOM 32-4
- adding memory 32-2
- ADDRESS RANGE switch (CPU) A-3
- ADDRESS RANGE switch (IOM) A-5
- ADDRESS STOP switch (CPU) 9-3
- admin.ec 3-12, B-1
- administrative ring 3-11, B-1
- AIM 14-1, 14-3, 14-5, 23-6, 23-7, 24-1, 24-4, 24-7, 24-9
- ALARM RESET button (IOM) 36-11
- ALARM switch (CPU) A-3

ALARM switch (IOM) A-5  
 ALARM switch (SCU) A-2  
 answering service 3-12, 13-1, 30-1, 39-1, B-1  
   failure 13-5  
   log 3-15, B-1  
   message 6-2, 7-1, 7-4  
   process 3-10, B-1  
 argument 4-2, B-1  
 assigning a disk drive 20-4, 21-5, B-1  
 ASSIGNMENT switch (CPU) 9-2, A-3  
 ASSIGNMENT switch (IOM) 9-1, A-5  
 attach command (initializer) 29-3, 29-4  
 attaching a channel 29-2  
 attended command (exec) 31-2  
 attended mode 31-1, B-1  
   setting 31-2  
 attended service 31-1, B-1  
 ATTN/RESET key (console) 5-5, 5-8, 11-6  
 audible alarm 5-5, 6-3, 7-1, 11-2, 14-2, 17-6,  
   19-1, 19-2, 20-1, 20-2, 21-3, 21-4,  
   21-8, 38-3, B-1, B-2  
 auth command (exec) 19-3, 20-3  
 authenticating a disk 20-3  
 authenticating a tape 19-3  
 authenticating a user I/O disk 20-3  
 auto command (exec) 31-2  
 auto exec\_com 3-12, 13-1, 16-1, 31-1, 35-1,  
   35-2, 37-2, B-2  
 AUTO switch (CPU) A-4  
 automatic mode 13-1, 31-1, 35-1, 36-2, 36-3,  
   36-5, 36-7, 36-10, B-2  
   setting 31-2  
 AUTOMATIC RATE SWITCH (IOM) 9-2  
 automatic reboot mode 31-1, B-2  
 automatic recovery failure 37-1  
 automatic recovery procedures 35-1, 35-2, 36-2,  
   36-3, 36-4, 36-5, 36-6, 36-7, 36-8,  
   36-10, 36-11

## B

backup 3-13, B-2  
   daemon 3-13, B-2  
   dumping 3-13  
   functions 3-13  
   hierarchy 3-13, B-9  
   recovery 3-13, B-13  
   reloading 3-13, B-14  
   retrieving 3-13, B-14  
   volume 3-13, B-18

backup commands 3-13, B-2  
   end\_dump 17-3  
   end\_volume\_dump 17-3  
 backup commands (cont.)  
   retrieve 27-4  
   retrieve\_from\_volume 27-1, 27-5  
   wakeup\_dump 17-3  
   wakeup\_volume\_dump 17-3  
 backup daemon message 6-2, 7-1, 7-3  
 backup dump 26-1, B-2  
 backup hierarchy dumping  
   see incremental hierarchy dumping  
 backup tape 26-1, B-2  
 backup volume dumping  
   see incremental volume dumping  
 banner bars 22-1, B-2  
 BCE 3-5, 3-6, 31-1, 33-1, 34-2, B-2  
   booting 3-6  
   bootload 3-6, 11-1, 11-8  
   bootloading 11-1  
   initializing 11-1  
   message 7-1, 7-2, B-2  
   partition 3-6, 11-14, B-2  
   returning after system failure 13-4, 35-2,  
     36-1, 36-3, 36-5, 36-7, 36-10, 36-12  
   shutting down 36-8, 36-11  
   states 3-6  
 BCE 24000 command (DMP/VIP) 34-7, 36-5,  
   36-10  
 BCE 24000 command (DPU) 34-7, 36-5, 36-10  
 BCE 24002 command (DMP/VIP) 34-7, 36-12  
 BCE 24002 command (DPU) 34-7, 36-12  
 bce command (BCE) 11-8, 16-1  
 BCE commands 3-6, 4-3, 5-5, 6-2, B-2  
   bce 11-8, 16-1  
   boot 13-1, 35-2, 37-2, 37-3  
   config 11-9, 12-1, 12-2, 12-3, 12-4, 12-5  
   die 36-8, 36-11  
   dump 35-1  
   esd 35-1, 37-1  
   fwload 11-11, 16-1, 36-12, 37-2  
   rpv 16-1  
 BCE/Multics system tape 3-6, 3-8, 11-1, B-2  
 beeper  
   see audible alarm  
 boot command (BCE) 13-1, 35-2, 37-2, 37-3  
 BOOT SOURCE switch (IOM) 9-2, 11-2, 11-13,  
   A-5  
 booting 3-5, B-2, B-3  
 booting BCE 3-6, 11-8  
   problems 11-13  
 booting Multics 3-5, 3-8, 13-1  
 BOOTLOAD button (console) 11-4, 11-13  
 BOOTLOAD button (IOM) 11-4



bootload command environment  
     see BCE  
 bootload console 3-6, 3-11, 5-1, 11-3, B-3  
     adding alternate 32-7  
     and MCA 5-6  
     changing 32-8  
     deleting 32-9  
 bootload CPU 3-5, 9-2, 32-2, 32-3, 33-1, 36-2,  
     36-3, 36-4, 36-5, 36-6, 36-7, 36-9,  
     36-10, B-3  
 bootload IOM 3-5, 9-1, 11-1, B-3  
 bootload processor  
     see bootload CPU  
 bootload SCU 3-5, 9-1, 32-4, B-3  
 bootload session 3-5, B-3  
 bootloading  
     see booting  
 bootloading BCE  
     see booting BCE  
 bootloading Multics  
     see booting Multics  
 bootloads  
     cold  
         see boots, cold  
     warm  
         see boots, warm  
 boots  
     cold 3-5, B-4  
     warm 3-5, B-19  
 BOT light (tape drive) 11-1, 11-3, 11-12, 19-4  
 box 2-1, B-3  
 bringing the system up 3-8, 16-1, A-1, B-3  
 bulk data input 23-3  
 bump command (initializer) 17-2, 29-2  
 bumping an absentee job 30-4

## C

CABINET SHUTDOWN button (CPU) 8-1  
 CABINET SHUTDOWN button (IOM) 8-1  
 CABINET SHUTDOWN button (SCU) 8-1  
 CACHE ENABLE switches (CPU) 13-4  
 canceling a storage system disk mount 21-4  
 canceling shutdown 17-6  
 CAPS LOCK key (DPU) 34-1  
 card decks  
     checking 23-3  
     control cards 23-4  
     format 23-3  
         bulk data input 23-3  
         remote job entry 23-3  
     preparing 23-4  
     reading 23-3  
 card punch 2-6, 23-1, 23-2, B-3

card reader 2-6, 23-7, B-3  
 CARRIAGE RETURN key (init. term.) 6-1  
 cat command (exec) 26-4  
 catchup dump 26-1, B-3  
 catchup hierarchy dumping  
     see consolidated hierarchy dumping  
 catchup tape 26-1, B-3  
 catchup volume dumping  
     see consolidated volume dumping  
 CCU 2-6, 23-6, B-4  
 central processing unit  
     see CPU  
 CF command (DMP/VIP) 34-6, 34-7  
 CF command (DPU) 34-6, 34-7  
 CFG command (DMP/VIP) 34-6, 34-7  
 CFG command (DPU) 34-6, 34-7  
 change\_wdir command (Multics) 27-3  
 changing bootload console 32-8  
 changing printer request type 22-5, 22-6  
 changing remote printer print formats 24-7  
 changing state of CPU  
     in BCE 12-1  
 changing state of FNP  
     in BCE 12-4  
 changing state of IOM  
     in BCE 12-3  
 changing state of SCU  
     in BCE 12-2  
 changing state of system console  
     in BCE 12-5  
 changing unit connected to DPU 34-4  
 CHANNEL NUMBER CODE switch (IOM) 9-2,  
     11-2, A-6  
 CHANNEL SELECT switch (IOM) 9-2, 11-2,  
     A-6  
 channels 29-4  
     attaching 29-3  
     communications 2-3, 24-8, B-4  
     detaching 29-3  
     IOM 2-1, 9-2, A-7, B-10  
     logical  
         adding 32-6  
         deleting 32-7  
         removing 29-3  
 checking a card deck 23-3  
 checking switch settings 9-1  
 clearing a store unit 32-2  
 clock 3-15, 11-9, 13-1, B-3  
     setting 11-9  
 clock card 3-9, 11-9, 11-14, B-3

- cold boot 3-5, B-4
- cold bootload
  - see cold boot
- combined card unit
  - see CCU
- commands 4-2, B-4
  - backup 3-13, B-2
  - BCE 3-6, 4-3, 5-5, 6-2, B-2
  - daemon 3-12, 4-3, 5-5, 6-2, B-6
  - exec 3-12, 4-3, 5-5, 6-2, B-7
  - getting help 4-4
  - I/O daemon 3-13, B-10
  - initializer 3-11, 4-3, 5-5, 6-2, B-9
  - Multics 4-2, 4-3, 5-5, 6-2, B-12.1
  - regular vs exec vs daemon 4-2
  - short form vs long form 4-2
- communications channel 2-3, 24-8, B-4
- communications line 2-3, 24-8, 24-13, 25-1, B-4
  - getting info 24-13
- communications processor 2-3, B-4
- comp command (exec) 26-6
- complete dump 26-1, 33-1, B-4
- complete dump tape 26-1, B-4
- complete hierarchy dumping 26-6
- complete volume dumping 26-7
- config cards 3-9
  - clock 3-9, 11-9, 11-14, B-3
  - cpu 2-1, 3-9, 12-1, B-5
  - iom 2-2, 3-9, 12-3, B-10
  - mem 2-1, 3-9, 12-2, B-12
  - prph 2-2, 3-9, 12-4, 12-5, A-7, B-13
- config command (BCE) 11-9, 12-1, 12-2, 12-3, 12-4, 12-5
- config deck 3-9, 3-16, 9-1, 12-1, 32-1, 33-1, 38-2, A-1, A-7, B-4, B-5
  - clock card 3-9, 11-9, 11-14, B-3
  - cpu card 3-9, 12-1, B-5
  - editing 12-1
  - iom card 3-9, 12-3, B-10
  - mem card 3-9, 12-2, B-12
  - prph card 3-9, 12-4, 12-5, A-7, B-13
- configuration 3-9, 33-1, B-5
- configuration card
  - see config card
- configuration deck
  - see config deck
- configuration panels
  - displaying 34-5
- console 5-1, B-5
  - audible alarm 5-5
  - bootload 3-6, 3-11, 5-1, 11-3, B-3
    - adding alternate 32-7
    - changing 32-8
    - deleting 32-9
  - changing state
    - in BCE 12-5
  - correcting errors 5-3
  - entering commands 5-3
- console (cont.)
  - input prompt 5-1
  - interrupting system 5-4
  - issuing commands 5-2, 5-5
  - messages 5-5
  - rerouting activity 5-5
  - system 5-1, B-16
  - typing input 5-1
- console buttons
  - BOOTLOAD 11-4, 11-13
  - CR 36-12
  - INITIALIZE 11-4, 11-13
  - OFFLINE 18-1
  - ONLINE 8-1
  - POWER OFF 18-1
  - POWER ON 8-1
- console keys
  - ATTN/RESET 5-5, 5-8, 11-6
  - EOM 5-4, 5-8, 11-7, 11-10
  - OPERATOR ERROR 5-3, 5-8
  - REQUEST 5-2, 5-4, 5-8
  - RETURN 5-2, 5-4, 5-8, 11-4, 11-7, 11-10, 11-13
- console switches
  - POWER 8-1, 18-1
- consolidated dump 26-1, B-5
- consolidated hierarchy dumping 26-4
- consolidated tape 26-1, B-5
- consolidated volume dumping 26-5
- control cards 23-4
- control file 27-3, B-5
- controllers 11-14
  - disk 2-5, 11-11, B-7
  - tape 2-5.1, 11-1, 11-7, 11-11, 36-12, 37-2, B-17
  - unit record 2-6, 11-11, 37-2, B-18
- converting a disk drive
  - from storage system to user I/O 20-5, 21-5
  - from user I/O to storage system 20-5, 21-5
- coordinator 14-1, 14-3, 22-1, B-5
  - logging in 14-3
  - problems 14-8
  - process 3-13, B-5
  - starting 14-3
- coordinator request number 22-1, 22-10, B-5
- copy command (daemon) 22-4
- CPU 2-1, B-3
  - adding 29-1, 32-1
  - bootload 3-5, 9-2, 32-2, 32-3, 33-1, 36-2, 36-3, 36-4, 36-5, 36-6, 36-7, 36-9, 36-10, B-3
  - changing state
    - in BCE 12-1
  - deleting 29-1, 32-2
  - initializing 11-3, 32-1
- CPU buttons
  - CABINET SHUTDOWN 8-1
  - EXECUTE 36-2, 36-3, 36-7
  - INIT-CLEAR 11-3, 32-1, 36-6, 36-9, 36-11
  - INITIALIZE & CLEAR 11-3, 32-1, 36-4,

CPU buttons (cont.)  
 36-7, 36-8  
 POWER OFF 18-1  
 POWER ON 8-1

cpu card 2-1, 3-9, 12-1, B-5

CPU switches  
 ADDRESS RANGE A-3  
 ADDRESS STOP 9-3  
 ALARM A-3  
 ASSIGNMENT 9-2, A-3  
 AUTO A-4  
 CACHE ENABLE 13-4  
 CYCLE 36-3, 36-4, 36-6, 36-8, A-4  
 DATA 36-3, 36-7, 36-12, A-4  
 ENABLE MATCH A-4  
 EXECUTE PB/SCOPE REPEAT 9-3, A-4  
 EXECUTE SWITCHES/EXECUTE FAULT  
 36-2, 36-3, 36-7, A-4  
 FAULT CONTROL 9-3, A-3  
 INITIALIZE & CLEAR/INITIALIZE  
 CONTROL 9-3, A-4  
 INITIALIZE ENABLE A-3  
 MAINTENANCE PANEL MODE A-3  
 OPERATING MODE 11-13, A-3  
 PORT ENABLE 9-2, 11-3, 32-1, 36-4, 36-6,  
 36-9, 36-11, A-3  
 PROCESSOR FAULT BASE ADDRESS A-3  
 PROCESSOR NUMBER" A-3  
 TEST CONTROL 9-3

CR button (console) 36-12

CR key (DPU) 34-1

crash 3-8, 31-1, 32-9, 33-1, 37-3, B-5  
 during startup 13-4  
 FNP 3-8, B-8  
 system 3-8, 35-1, 35-2, B-16.1

cross barred 2-3, 2-5, 38-1, B-5

cross retrieval 27-3, B-5

crt terminal 2-5, B-5

CSD 1-3, 9-2, 9-3, 11-15, 19-6, 19-8, 34-6,  
 36-12, A-8, B-6

CSU6004 5-1, 8-1, 11-4, 18-1, B-6

CSU6601 5-1, 8-1, 11-3, 18-1, 36-12, B-5

customer services division  
 see CSD

CYCLE switch (CPU) 36-3, 36-4, 36-6, 36-8,  
 A-4

## D

daemon commands 3-12, 4-3, 5-5, 6-2, B-6  
 copy 22-4  
 defer 22-11  
 go 14-2, 14-5, 14-6, 16-1, 22-2, 22-6, 22-7,  
 22-9, 22-10, 22-11, 22-12, 23-1,  
 23-3, 24-4, 24-6, 24-7, 24-9, 24-11,  
 25-3  
 halt 17-3, 22-5, 24-7  
 kill 22-9  
 list 14-8, 22-8, 24-12  
 list\_daemon\_requests 22-9, 23-2

daemon commands (cont.)  
 new\_device 22-5  
 next 22-11  
 print 22-4  
 print\_devices 14-6, 22-7, 24-12  
 print\_line\_ids 24-13  
 quit 22-9, 23-8  
 read\_cards 23-7, 23-8  
 ready 24-7  
 reinit 22-12  
 release 23-8  
 restart 22-3, 22-10  
 sample\_hs 14-2, 16-1, 22-4, 22-6  
 shutdown 40-1  
 start 22-13  
 station 25-2  
 status 22-6

daemons 3-12, 3-16, B-6  
 backup 3-13, B-2  
 Data Management 40-1, B-6  
 logging in 40-1  
 dumper  
 logging in 15-1, 26-2, 26-4, 26-6  
 logging out 26-10  
 starting 15-1  
 I/O 3-13, B-10  
 logging in 14-1  
 logging out 22-13, 23-8  
 starting 14-1  
 retrieval  
 logging in 27-1, 27-2  
 logging out 27-5

data base 3-1, 3-5, B-6

Data Management 40-1, B-6

Data Management daemon 40-1, B-6

Data Management system  
 shutdown problems 40-2  
 shutting down 40-1

DATA switches (CPU) 36-3, 36-7, 36-12, A-4

datanet 2-3, B-6

daylight savings time 11-9

defer command (daemon) 22-11

deferring a printer request 22-10

deferring an absentee job 30-5

DEL key (DMP/VIP) 34-6, 34-7

DEL key (DPU) 34-1, 34-6, 34-7

del\_lv command (initializer) 13-3, 21-4, 21-7

deleting a CPU 29-1, 32-2

deleting a disk drive 20-6, 21-6

deleting a FNP 32-6

deleting a logical channel 32-7

deleting a SCU 29-1, 32-4

deleting a tape drive 19-7

deleting an IOM 32-5

- deleting bootload console 32-9
- deleting memory 32-4
- density 2-5.1, 11-1, 11-12, 26-1, 26-2, 26-4, 26-6, B-6
- deny command (exec) 19-3, 20-4
- denying a tape 19-3
- denying a user I/O disk 20-4
- detach command (initializer) 29-3
- detaching a channel 29-3
- device class 14-1, 14-4, 14-5, 22-13, 24-2, 24-5, 24-10, 24-13, B-6
  - getting info 22-7, 24-12
- devices
  - FIPS-style B-9, B-15
  - getting info 22-7, 24-12
  - head-of-string B-9, B-15
  - logical 24-1, B-11
  - major 24-1, 25-2, B-12
  - minor 24-1, 24-2, 24-5, 24-10, 25-2, B-12.1
  - remote 24-1, B-14
    - logging in 24-7, 24-9
  - storage 2-5, B-15
  - unit record 2-5.2, B-18
- DIA 2-3, B-6
- DIA board A-7
- diagnostics processor unit
  - see DPU
- dial command (Multics) 5-5, 6-3, 25-1
- die command (BCE) 36-8, 36-11
- direct interface adapter
  - see DIA
- directory 3-1, 3-16, B-6
  - hierarchy 3-1, B-6
  - salvager 3-14, B-6
- disk controller 2-5.1, 11-11, B-7
- disk devices
  - FIPS B-15
- disk drive
  - 451 3-3
  - 500 3-3
  - 501 3-3
  - 3380 3-3
  - 3390 3-3
- disk drive buttons
  - POWER OFF 18-1
  - POWER ON 8-1
  - READY 38-3
  - START 20-2, 20-4, 21-2, 21-3, 21-4
  - STOP 18-1
  - SUBSYSTEM POWER OFF 18-1
  - SUBSYSTEM POWER ON 8-1
  - UNIT EMERGENCY 8-1
- disk drive lights
  - READY 21-4
- disk drive switches
  - ONLINE/OFFLINE 18-1
  - START/STOP 8-1, 18-1
- disk drives 2-5, B-7
  - 451 2-5, 8-1, 18-1, 20-1, 21-2, 21-3
    - moving 38-3
  - 500 2-5, 8-1, 18-1, 20-1, 21-2, 21-3
  - 501 2-5, 8-1, 18-1, 20-1, 21-2, 21-3
  - 3300 8-1
  - 3380 2-5.1, 18-1
  - 3381 2-5.1, 18-1
    - adding 20-5, 21-5
    - converting 20-5, 21-5
    - deleting 20-6, 21-6
    - readying 20-4, 21-4
- disk error message 5-5, 7-1, 7-2
- disk pack 2-5, 33-1, B-7
- disks 2-5, B-6
  - assigning 20-5, 21-5, B-1
  - authenticating 20-3
  - canceling mount 21-4
  - denying 20-4
  - getting info 20-6, 21-8
  - killing mount request 19-3, 20-4, B-11
  - mounting 20-1, 21-1, 21-3
  - responding to problems 38-3
  - storage system 2-5, B-16
  - user I/O 2-5, B-18
- displaying configuration panels 34-5
- distributed processing system 8
  - see DPS 8
- DMP 2-8, B-7
- DMP/VIP 36-4, 36-9
  - getting connected 34-5
  - modes 34-5
    - VIP 34-5, B-18
  - naming conventions 34-4
  - typing conventions 34-4
- DMP/VIP commands
  - BCE 24000 34-7, 36-5, 36-10
  - BCE 24002 34-7, 36-12
  - CF 34-6, 34-7
  - CFG 34-6, 34-7
  - ST CU 34-7, 36-5, 36-10
- DMP/VIP keys
  - <F1> 34-6, 34-7
  - DEL 34-6, 34-7
- down command (initializer) 17-1
- dprint 22-1, B-7
- DPS 8 2-6, B-7
- DPU 2-6, 36-4, 36-9, B-6
  - changing unit connected 34-4
  - finding out which unit is connected 34-3
  - getting help with commands 34-4
  - modes 34-1
    - ready 34-1, B-13
    - TM 34-2, B-17
    - VIP 34-2, 34-5, 36-4, 36-9, B-18
  - naming conventions 34-1
  - typing conventions 34-1
  - use of carriage return 34-1

DPU commands  
 U 34-3, 34-6  
 ? 34-4, 34-6  
 BCE 24000 34-7, 36-5, 36-10  
 BCE 24002 34-7, 36-12  
 CF 34-6, 34-7  
 CFG 34-6, 34-7  
 OFL 34-2, 34-4, 34-6  
 QUIT 34-3, 34-6  
 ST CU 34-7, 36-5, 36-10  
 SUSP 34-6  
 TM 34-3, 34-7  
 VIP 34-3, 34-6

DPU keys  
 <BRK> 34-3, 34-7  
 <F1> 34-6, 34-7  
 CAPS LOCK 34-1  
 CR 34-1  
 DEL 34-1, 34-6, 34-7  
 RETURN 34-1

driver B-7  
 printer 14-1, 22-1  
 logging in 14-4  
 logging out 22-13  
 starting 14-4, 22-2  
 problems 14-8  
 punch 14-5, 23-1  
 logging in 14-5  
 logging out 23-6, 23-8  
 restarting 23-2  
 starting 14-5, 23-1  
 stopping 23-2  
 reader  
 logging in 23-6  
 logging out 23-7, 23-8  
 starting 23-7  
 stopping 23-8

driver process 3-13, B-7

drives  
 disk 2-5, B-7  
 tape 2-5, B-17

drop command (initializer) 6-4

dump 3-8, 31-1, 33-1, 36-2, 36-3, 36-4, 36-5,  
 36-6, 36-7, 36-8, 36-10, 36-11, 37-1

dump command (BCE) 35-1

dump control file 26-6

dump exec\_com 3-12, 37-1

DUMP partition 37-1

dump\_mpx command (initializer) 39-1

dumper daemons  
 logging in 15-1, 15-3, 26-2, 26-4, 26-6  
 logging out 26-8, 26-9  
 starting 15-1

dumping 3-13, B-7  
 complete hierarchy 26-7  
 complete volume 26-7  
 consolidated hierarchy 26-4  
 consolidated volume 26-5  
 incremental hierarchy 15-1, 17-3, 26-2  
 incremental volume 15-2, 15-3, 17-3, 26-3

dumping a FNP 39-1

dumping Multics 13-4, 37-1

dumps  
 backup 26-1, B-2  
 catchup 26-1, B-3  
 complete 26-1, 33-1, B-4  
 consolidated 26-1, B-5  
 incremental 26-1, B-9  
 wakeup 26-1, B-19

dynamic maintenance panel  
 see DMP

dynamic reconfiguration 3-9, 32-1, B-7

**E**

editing clock card 11-9

editing the config deck 12-1

emergency shutdown  
 see ESD

ENABLE MATCH switch (CPU) A-4

end\_dump command (backup) 17-3

end\_volume\_dump command (backup) 17-3

EOM key (console) 5-4, 5-8, 11-7, 11-10

erase character (#) 5-3, 6-1, B-7

error message documentation 7-4

ESD 3-8, 13-4, 17-6, 31-1, 36-2, 36-3, 36-4,  
 36-5, 36-6, 36-7, 36-8, 36-10,  
 36-11, 37-1, 38-3, B-7  
 failure 37-3  
 forcing 36-12

esd command (BCE) 35-1, 37-1

exec commands 3-12, 4-3, 5-5, 6-2, B-7  
 attended 31-3  
 auth 19-3, 20-3  
 auto 31-2  
 cat 26-4  
 comp 26-6  
 deny 19-3, 20-4  
 inc 15-1, 16-1, 26-2  
 io 14-1, 16-2  
 iol 14-1  
 prt 22-2  
 puna 23-1  
 punch\_end 23-2  
 punch\_restart 23-2  
 reprint 22-5  
 reset\_tabs 6-5  
 unattended 31-1  
 vcomp 26-7  
 vcons 26-5  
 vinc 15-2, 16-1, 26-3

exec\_coms 3-12, B-7  
 admin.ec 3-12, B-1  
 auto 3-12, 13-1, 16-1, 31-1, 35-1, 35-2,  
 37-2, B-2  
 dump 3-12, 37-1  
 go 3-12  
 rtb 3-12, 35-2, 36-2, 36-3, 36-5, 36-7, 36-10  
 system\_start\_up.ec 3-12, 14-1, 15-1, 26-2,  
 B-17

execute 2-1

EXECUTE button (CPU) 36-2, 36-3, 36-7  
 EXECUTE PB/SCOPE REPEAT switch (CPU)  
 9-3, A-4  
 EXECUTE SWITCHES/EXECUTE FAULT  
 switch (CPU) 36-2, 36-3, 36-7, A-4  
 executing fault 36-2, 38-3, B-7  
 executing switches 36-2, B-8  
 executing switches (DPS 8) 36-4  
 with different bootload CPU 36-9  
 executing switches (L68) 36-3  
 with different bootload CPU 36-6

## F

failures  
 answering service 13-4  
 automatic recovery 37-1  
 different kinds 35-1, 36-1  
 ESD 37-3  
 FNP 39-1  
 hardware 33-1, 36-12  
 MPC 38-1  
 Multics dump 37-3  
 Multics reboot 37-3  
 recording 37-3  
 startup 13-4  
 system crash 3-8, B-16.1  
 recognizing 35-1, 35-2  
 responding 35-1, 35-2  
 system hang B-9  
 recognizing 36-1  
 system loop B-16.1  
 recognizing 36-1  
 FAULT CONTROL switch (CPU) 9-3, A-3  
 finding out how many users are logged in 29-1  
 finding out which backup daemons are logged in  
 15-3  
 finding out which I/O daemons are logged in  
 14-8  
 finding out which unit is connected to DPU  
 34-3  
 finding out which users are logged in 29-1  
 FIPS channel  
 IMU B-15  
 FIPS disk devices B-15  
 FIPS-style devices B-9, B-15  
 firmware 2-1, 3-15, 11-1, 11-15, B-8  
 loading 11-7, 11-11  
 reinitializing 38-1, 38-2  
 reloading 37-2, 38-1  
 firmware lights 38-1, B-8  
 fixing printer paper 11-12, 22-2  
 FNP 2-3, 29-4, B-8  
 adding 32-5  
 changing state  
 in BCE 12-4  
 crash 3-8, B-8  
 deleting 32-6

FNP (cont.)  
 dumping 39-1  
 failure 39-1  
 loading 13-2, 32-5  
 reloading 39-2  
 shutting down 39-2  
 taking out of service 39-1  
 FNP buttons  
 POWER OFF 18-1  
 POWER ON 8-1  
 FNP switches  
 6000 EMERGENCY A-7  
 6000 MAILBOX A-7  
 6000 TERMINATE A-7  
 HNP MAILBOX A-7  
 HNP SPECIAL INTERRUPT A-7  
 HNP TERMINATE A-7

forcing an absentee job to log in 30-6  
 forcing an ESD 36-12  
 foreign tapes 19-1  
 front-end 2-3, B-8  
 front-end network processor  
 see FNP  
 fwload command (BCE) 11-11, 16-1, 36-12,  
 37-2

## G

getting connected to DMP/VIP 34-5  
 getting help with commands 4-4  
 getting help with DPU commands 34-4  
 getting info about absentee jobs 30-7  
 getting info about communications lines 24-13  
 getting info about device classes 14-6, 22-7,  
 24-12  
 getting info about devices 14-6, 22-7, 24-12  
 getting info about disk mounts 20-6, 21-8  
 getting info about printer requests 22-9  
 getting info about punch requests 23-2  
 getting info about request types 14-6, 22-7,  
 24-12  
 getting info about tape mounts 19-7  
 go command (daemon) 14-2, 14-5, 14-6, 16-1,  
 22-2, 22-6, 22-7, 22-9, 22-10,  
 22-11, 22-12, 23-1, 23-3, 24-4,  
 24-6, 24-7, 24-9, 24-11, 25-3  
 go exec\_com 3-12

## H

halt command (daemon) 17-3, 22-5, 24-7  
 HALTED light (MPC) 11-15  
 hang 36-1  
 hardcopy terminal 2-5, B-8

hardcore 3-9, B-8  
 hardcore message 3-15, B-8  
 hardcore partition 3-9, B-8  
 hardcore supervisor 3-9, B-8  
 hardware 2-1, B-8  
     failure 33-1, 36-12  
 hardwired 5-1, 25-1, B-8  
 head-of-string device B-8, B-15  
 headsheet 22-1, B-9  
 help command (initializer) 4-4  
 help file 1-3, B-9  
 hierarchy  
     directory 3-1, B-6  
     storage system 3-1, B-16  
 hierarchy backup 3-13, B-9  
 hierarchy dump maps 26-8  
 hierarchy dump tapes 33-1  
 hierarchy dumping  
     complete 26-6  
     consolidated 26-4  
     incremental 15-1, 17-3, 26-2  
 hierarchy retrievals 27-2  
 high-speed printer 2-5.2, B-9  
 hmu command (initializer) 17-2, 29-1, 30-2  
 HNP MAILBOX switch (FNP) A-7  
 HNP SPECIAL INTERRUPT switch (FNP) A-7  
 HNP TERMINATE switch (FNP) A-7  
  
     I  
  
 I/O daemon commands 3-13, B-10  
 I/O daemon message 6-2, 7-1, 7-3  
 I/O daemons 3-13, B-10  
     logging in 14-1  
     logging out 22-13, 23-8  
     starting 14-1  
 iboot command (MCA) 11-5, 11-12  
 IMU 2-2, B-9, B-12, B-15  
     adding 32-4  
     FIPS channel B-15  
     initializing 32-4  
     IPC 2-2, 2-5.1, 11-8, B-10  
     MCA 2-2, 5-6, 11-4, 11-5, 11-6, B-12  
 inactive state 12-1, B-9  
 inc command (exec) 15-1, 16-1, 26-2  
 incremental dump 26-1, B-9  
 incremental hierarchy dumping 15-1, 17-3, 26-2  
 incremental tape 26-1, 37-2, B-9  
 incremental volume dumping 15-2, 17-3, 26-3

information multiplexer unit  
     see IMU  
 init command (MCA) 32-4  
 INIT-CLEAR button (CPU) 11-3, 32-1, 36-6,  
     36-9, 36-11  
 init. term. keys  
     CARRIAGE RETURN 6-1  
 initialization 3-8, B-9  
 INITIALIZE & CLEAR button (CPU) 11-3,  
     32-1, 36-4, 36-7, 36-8  
 INITIALIZE & CLEAR/INITIALIZE CONTROL  
     switch (CPU) 9-3, A-4  
 INITIALIZE button (console) 11-4, 11-13  
 INITIALIZE button (IOM) 32-4  
 INITIALIZE button (MPC) 8-1, 11-15, 38-3  
 INITIALIZE button (SCU) 11-13, 32-3  
 INITIALIZE ENABLE switch (CPU) A-3  
 INITIALIZE ENABLE switch (IOM) A-5  
 INITIALIZE switch (IOM) 36-12  
 initializer 3-10, 3-16, B-9  
     command response 7-1, 7-4, B-9  
     hang 36-1, B-9  
     message 6-2  
     process 3-10, 3-16, 6-2, 13-1, 36-1, B-10  
 initializer commands 3-11, 4-3, 5-5, 6-2, B-9  
     abs  
         bump 30-4  
         defer 30-5  
         list 30-7  
         maxq 30-4  
         maxu 30-2  
         move 30-3  
         release 30-6  
         run 30-6  
         start 30-1  
         start queue 30-3  
         stop 17-4, 30-1  
         stop queue 30-2  
         suspend 30-5  
     accept 5-6, 6-3  
     add\_lv 21-1, 21-3  
     add\_vol 21-2, 21-4  
     attach 29-3, 29-4  
     bump 17-2, 29-2  
     del\_lv 13-3, 21-4, 21-6  
     detach 29-3  
     down 17-1  
     drop 6-4  
     dump\_mpx 39-1  
     help 4-4  
     hmu 17-2, 29-1, 30-2  
     list\_disks 21-8  
     list\_vols 21-6  
     listen 24-8  
     load\_mpx 13-2, 32-5, 39-2  
     login 14-3, 14-4, 14-5, 15-2, 23-6, 24-2,  
         24-4, 24-7, 24-9, 26-2, 26-4, 26-6,  
         27-1, 40-1  
     logout 17-4, 22-13, 23-6, 23-7, 23-8, 26-8,  
         26-9, 27-5

initializer commands (cont.)

- maxu 29-2
- preload 19-6
- quit 22-3, 22-11
- rcp 19-7, 20-6, 21-8
- reconfigure 19-6, 19-7, 19-8, 20-5, 20-6,  
21-6, 21-7, 32-2, 32-3, 32-4, 32-4.1,  
32-5, 32-6, 32-7
- remove 6-4, 29-3
- set\_drive\_usage 20-5, 21-5, 21-6
- shutdown 17-4
- shutdown\_mpx 39-2
- sign\_off 4-4
- sign\_on 4-3, 16-1
- startup 13-1, 13-3
- substv 5-6, 6-4
- unload 19-5
- vacate\_pdir volume 21-6
- warn 19-8, 28-1, 28-2, 39-1, 39-3
- who 14-8, 15-3, 17-2, 29-1
- word 28-1

initializer terminal 3-11, 6-1, B-10

- adding terminal 6-3
- correcting errors 6-1
- dropping terminal 6-4
- entering commands 6-1
- interrupting system 6-1
- issuing commands 6-1, 6-2
- messages 6-2
- multiple terminals 6-2
- putting tabs back 6-5
- rerouting activity 6-4

initializing a CPU 11-3, 32-1

- DPS 8 36-6, 36-9, 36-11
- L68 36-4, 36-7, 36-8

initializing an IMU 32-4

initializing an IOM 32-4

initializing BCE 11-1

- problems 11-12

input mode 5-1, B-10

input/output multiplexer

- see IOM

integrated peripheral controller

- see IPC

interactive process 3-10, 3-16, B-10

INTERLACE switch (SCU) A-2

INTERRUPT BASE ADDRESS switch (IOM)

- 9-2, A-5

io command (exec) 14-1, 16-1

IO Daemon log 3-15, 22-1, B-10

iol command (exec) 14-1

IOM 2-1, B-10

- adding 32-4
- bootload 3-5, 9-1, 11-1, B-3
- changing state  
in BCE 12-3
- deleting 32-5
- initializing 32-5

IOM alarm

- responding 36-11

IOM BASE ADDRESS switch (IOM) 9-2, A-5

IOM buttons

- ALARM RESET 36-12
- BOOTLOAD 11-4
- CABINET SHUTDOWN 8-1
- INITIALIZE 32-4
- IOM INITIALIZE MANUAL 36-12
- MANUAL 32-4
- POWER OFF 18-1
- POWER ON 8-1
- SYSTEM INIT 11-4
- SYSTEM INITIALIZE 11-4

iom card 2-2, 3-9, 12-3, B-10

IOM channel 2-1, 9-2, A-7, B-10

IOM INITIALIZE MANUAL button (IOM)

- 36-12

IOM INITIALIZE switch (IOM) 32-4

IOM NUMBER switch (IOM) A-5

IOM switch (IOM) A-5

IOM switches

- ADDRESS RANGE A-5
- ALARM A-5
- ASSIGNMENT 9-1, A-5
- AUTOMATIC RATE 9-2
- BOOT SOURCE 9-2, 11-2, 11-13, A-5
- CHANNEL NUMBER CODE 9-2, 11-2, A-6
- CHANNEL SELECT 9-2, 11-2, A-6
- INITIALIZE 36-12
- INITIALIZE ENABLE A-5
- INTERRUPT BASE ADDRESS 9-2, A-5
- IOM A-5
- IOM BASE ADDRESS 9-2, A-5
- IOM INITIALIZE 32-4
- IOM NUMBER A-5
- IOM TEST/NORMAL 36-12
- MAINTENANCE PANEL MODE 36-12, A-5
- OPERATING MODE 9-2, 11-13, A-6
- OPERATION SELECTOR 9-2
- PORT ENABLE 9-1, A-5
- SCU PORT NUMBER 11-2, A-6
- SOURCE 9-2, 11-2, 11-13, A-5
- STOP CONDITIONS 9-2
- STORE CONTROL 9-2
- ZERO BASE S.C. PORT NO. 11-2, A-6

IOM TEST/NORMAL switch (IOM) 36-12

IPC 2-2, 2-5.1, 11-8, B-10

K

keyname 34-1, B-10

keypunch machine 2-6, B-10

kill character (@) 5-3, 6-1, B-11

kill command (daemon) 22-9

killing a mount request 19-3, 20-4, B-11

killing a printer request 22-9

kinds of failures 35-1, 36-1

L

label 3-13, B-11



- leaving note in system log book 33-1
- Level 68 2-6, B-11
- line printer 2-5.2, B-11
- link adapter 2-3, 9-2, 11-1, 11-12, 11-15, B-11
- list command (daemon) 14-8, 22-8, 24-12
- list\_daemon\_requests command (daemon) 22-9, 23-2
- list\_disks command (initializer) 21-8
- list\_vols command (initializer) 21-6
- listen command (initializer) 24-8
- LOAD button (tape drive) 11-1, 19-4
- load control group 3-15, 29-3, B-11
- LOAD/REWIND button (tape drive) 19-4
- load\_mpc command (Multics) 38-2
- load\_mpx command (initializer) 13-2, 32-5, 39-2
- loading a FNP 13-2, 32-5
- loading firmware 11-6, 11-11
  - problems 11-13, 11-14
- lock mode 5-2, B-11
- LOG partition 3-15, B-11
- logging 3-15, B-11
- logging in a remote device 24-7, 24-9
- logging in coordinator 14-3
- logging in Data Management daemon 40-1
- logging in dumper daemon 15-1, 15-3, 26-2, 26-4, 26-6
- logging in I/O daemons 14-1
- logging in printer driver 14-4
- logging in punch driver 14-5
- logging in reader driver 23-6
- logging in remote printer 24-2, 24-4
- logging in retrieval daemon 27-1, 27-2
- logging out a user 17-2, 29-2
- logging out daemons 17-4
- logging out dumper daemon 26-8, 26-9
- logging out I/O daemons 22-13, 23-8
- logging out printer driver 22-13
- logging out punch driver 23-6, 23-8
- logging out reader driver 23-7, 23-8
- logging out retrieval daemon 27-5
- logical channels
  - adding 32-6
  - deleting 32-7
- logical device 24-1, B-11

- logical volumes 2-5, 3-1, 13-3, B-11
  - private 3-3, B-13
    - mounting 21-3
  - public 3-3, B-13
    - mounting 21-1
  - root 3-5, 11-3, 13-3, B-15
- login command (initializer) 14-3, 14-4, 14-5, 15-2, 23-6, 24-2, 24-4, 24-7, 24-9, 26-2, 26-4, 26-6, 27-1, 40-1
- login command (Multics) 5-7, 27-2, 32-7, 32-8, 32-9, 38-1
- login message 7-1, 7-3
- logout command (initializer) 17-4, 22-13, 23-6, 23-7, 23-8, 26-8, 26-9, 27-5
- logout command (Multics) 5-8, 27-4, 32-8, 32-9, 32-10, 38-2
- logout message 7-1, 7-3
- logs
  - answering service 3-15, B-1
  - IO Daemon 3-15, 22-1, B-10
  - syserr 3-15, B-16
- loop 36-1
- low-order memory 3-5, 32-4, 36-12, B-11
- LWR STORE SIZE switch (SCU) A-2
- LWR STORE switch (SCU) A-2

**M**

- magnetic tape processor
  - see MTP
- maintenance channel adapter B-12
  - see MCA
- MAINTENANCE PANEL MODE switch (CPU) A-3
- MAINTENANCE PANEL MODE switch (IOM) 36-12, A-5
- major device 24-1, 25-2, B-12
- MANUAL button (IOM) 32-4
- manual mode 31-1, 35-2, 36-2, 36-3, 36-5, 36-7, 36-10, B-12
  - setting 31-4
- MARGIN CONTROL MODE switch (SCU) A-2
- MASK A MASK/PORT ASSIGNMENT switch (SCU) 9-1, 32-3, 36-6, 36-8, 36-9, 36-11
- MASK/PORT ASSIGNMENT switch (SCU) A-2
- mass storage processor
  - see MSP
- master console
  - multidrop interface to IMUs B-12
- master directory 3-3, B-12
- match id 22-9, 22-11, B-12
- maxu command (initializer) 29-2

MCA 2-2, 5-6, 11-4, 11-5, 11-6, B-12  
  and bootload console 5-6  
  communicating with 5-6  
MCA commands  
  iboot 11-5, 11-12  
  init 32-3  
  rload 32-4  
MDI B-12  
mem card 2-1, 3-9, 12-2, B-12  
MEM CLEAR button (SCU) 32-3  
memory 2-1, B-12  
  adding 32-2  
  deleting 32-4  
MEMORY SELECT switch (SCU) 32-3  
message coordinator 3-11, B-12  
message coordinator id 3-13  
message coordinator message 7-1, 7-2, 7-3, B-12  
message coordinator name 3-13, B-12  
message coordinator source id B-12  
message coordinator terminal 6-1, B-12.1  
messages  
  answering service 6-2, 7-1, 7-4  
  backup daemon 6-2, 7-1, 7-3  
  BCE 7-1, 7-2, B-2  
  disk error 5-5, 7-1, 7-2  
  hardcore 3-15, B-8  
  I/O daemon 6-2, 7-1, 7-3  
  initializer 6-2  
  login 7-1, 7-3  
  logout 7-1, 7-3  
  message coordinator 7-1, 7-2, 7-3, B-12  
  RCP 5-5, 7-1, 7-2  
  ready 5-2, 6-1  
  salvager 5-5, 7-1, 7-3  
  syserr 3-15, 7-1, 7-2, B-16  
  system 7-1  
  user 6-2  
microprogram 3-15, B-12.1  
microprogrammed peripheral controller  
  see MPC  
minor device 24-1, 24-2, 24-5, 24-10, 25-2,  
  B-12.1  
MODE switch (SCU) 32-3, 36-6, 36-8, 36-9,  
  36-11, A-2  
modes  
  attended 31-1, B-1  
  automatic 13-1, 31-1, 35-1, 36-2, 36-3, 36-5,  
  36-7, 36-10, B-1  
  automatic reboot 31-1, B-2  
  input 5-1, B-10  
  lock 5-2, B-11  
  manual 31-1, 35-2, 36-2, 36-3, 36-5, 36-7,  
  36-10, B-12  
  ready (DPU) 34-1, B-13  
  TM (DPU) 34-1, B-18  
  unattended 19-5, 31-1, 35-1, B-18  
  unlock 5-2, B-18  
modes (cont.)  
  VIP (DMP/VIP) 34-5, B-18  
  VIP (DPU) 34-2, 34-5, 36-4, 36-9, B-18  
module 2-1  
mounted state 3-3, B-12.1  
mounting a private logical volume 21-3  
mounting a public logical volume 21-1  
mounting a tape 19-1  
mounting a user I/O disk 20-1  
moving 451 packs 38-3  
moving an absentee job 30-3  
MPC 2-3, B-12.1  
  failure 38-1  
MPC buttons  
  INITIALIZE 8-1, 11-15, 38-3  
  OPERATOR INTERRUPT 20-2  
  POWER OFF 18-1  
  POWER ON 8-1  
  RESET & BRANCH 38-3  
  START 8-1, 11-15  
MPC lights  
  firmware 38-1  
MSP 2-5.1, B-12  
MTP 2-5.1, 11-7, 37-2, B-11  
Multics 3-8, B-12.1  
  booting 3-5, 3-8, 13-1  
  bootload 3-5, 3-8, 13-1  
  dump failure 37-3  
  dumping 13-4, 37-1  
  reboot failure 37-3  
  rebooting 37-2  
Multics commands 4-2, 4-3, 5-5, 6-2, B-12.1  
  change\_wdir 27-3  
  dial 5-5, 6-3, 25-1  
  load\_mpc 38-2  
  login 5-7, 27-2, 32-7, 32-8, 32-9, 38-1  
  logout 5-8, 27-4, 32-8, 32-9, 32-10, 38-2  
  qedx 27-3  
  set\_system\_console 5-8, 32-8, 32-10  
multidrop interface to IMUs  
  master console B-12  
multiplexer 2-3, B-12.1  
  
N  
network 2-3, B-12.1  
new\_device command (daemon) 22-5  
next command (daemon) 22-11  
nonscheduled shutdown 17-5, 28-2  
  
O  
OFFLINE button (console) 18-1  
OFL command (DPU) 34-2, 34-4, 34-6  
ONLINE button (console) 8-1

- online software 3-9, B-12.1
- ONLINE/OFFLINE switch (disk drive) 18-1
- OPERATING MODE switch (CPU) 11-13, A-3
- OPERATING MODE switch (IOM) 9-2, 11-13, A-6
- operating system 3-5, B-12.1
- OPERATION SELECTOR switch (IOM) 9-2
- operations error report form 35-1, 35-2, 37-3
- operator authentication 4-3, 5-1, 5-2, 6-1, 13-2
- operator console 5-1, B-12.1
- OPERATOR ERROR key (console) 5-3, 5-8
- OPERATOR INTERRUPT button (MPC) 20-2
- operator's responsibilities 4-1
- orderly shutdown 3-8, B-12.1

## P

- paging mechanism A-8
- panel 2-6, B-13
- PANEL CLEAR button (SCU) 11-13, 32-3
- partitions
  - BCE 3-6, 11-14, B-2
  - DUMP 37-1
  - hardcore 3-9, B-8
  - LOG 3-15, B-11
- pathname 3-1, B-13
- performing ESD 37-1
- peripheral 2-1, 2-5, B-13
- physical volumes 2-5, 3-1, B-13
  - root 3-5, 11-3, 37-3, B-15
- placing a CPU in step
  - DPS 8 36-4, 36-5, 36-9
  - L68 36-3, 36-6
- PORT ENABLE switch (CPU) 9-2, 11-3, 32-1, 36-4, 36-6, 36-9, 36-11, A-3
- PORT ENABLE switch (IOM) 9-1, A-5
- PORT ENABLE switch (SCU) 9-1, 32-2, 36-6, 36-8, 36-9, 36-11, A-2
- POWER OFF button (SCU) 18-1
- POWER OFF button (console) 18-1
- POWER OFF button (CPU) 18-1
- POWER OFF button (disk drive) 18-1
- POWER OFF button (FNP) 18-1
- POWER OFF button (IOM) 18-1
- POWER OFF button (MPC) 18-1
- POWER OFF button (printer) 18-1
- POWER OFF button (tape drive) 18-1
- POWER ON button (console) 8-1

- POWER ON button (CPU) 8-1
- POWER ON button (disk drive) 8-1
- POWER ON button (FNP) 8-1
- POWER ON button (IOM) 8-1
- POWER ON button (MPC) 8-1
- POWER ON button (printer) 8-1
- POWER ON button (SCU) 8-1
- POWER ON button (tape drive) 8-1
- POWER switch (console) 8-1, 18-1
- powering off machines 18-1
- powering on machines 8-1
- preload command (initializer) 19-6
- preloading a tape 19-5, 31-1
- preparing a card deck 23-4
- print command (daemon) 22-4
- print\_devices command (daemon) 14-6, 22-7, 24-12
- print\_line\_ids command (daemon) 24-13
- printer buttons
  - POWER OFF 18-1
  - POWER ON 8-1
  - SKIP 11-12, 22-3
  - START 11-12, 22-3
- printer driver 14-1, 22-1
  - logging in 14-4
  - logging out 22-13
  - starting 14-4, 22-2
- printer switches
  - PUSH TO CLUTCH/TRACTORS/PUSH TO DECLUTCH 11-12, 22-3
- printers 2-5.2, 14-1, B-13
  - changing request types 22-5, 22-6
  - deferring requests 22-10
  - fixing the paper 11-11, 22-2
  - getting info 22-9
  - getting status 22-6
  - high-speed 2-5.2, B-9
  - killing requests 22-9
  - line 2-5.2, B-11
  - logging in 14-3
  - logging out 22-13
  - reinitializing 22-12
  - remote
    - changing print formats 24-7
    - logging in 24-2, 24-4
    - starting 24-2, 24-4
  - reprinting segment 22-5
  - reprocessing requests 22-10, 22-11
  - starting 14-3, 22-2
  - stopping 17-3
- printing terminal 2-5, B-13
- private logical volumes 3-3, B-13
  - mounting 21-3
- process directory 13-3, 21-6, B-13

processes 3-10, A-8, B-13  
   absentee 3-10, 3-16, B-1  
   answering service 3-10, B-1  
   coordinator 3-13, B-7  
   driver 3-13, B-7  
   initializer 3-10, 3-16, 6-2, 13-1, 36-1, B-10  
   interactive 3-10, 3-16, B-10  
   remote driver 24-1, B-14  
 processor 2-1, B-13  
 PROCESSOR FAULT BASE ADDRESS switch  
   (CPU) A-3  
 PROCESSOR NUMBER switch (CPU) A-3  
 program 2-1, B-13  
 prph card 2-2, 3-9, 12-4, 12-5, A-7, B-13  
 prta command (exec) 22-2  
 public logical volumes 3-3, B-13  
   mounting 21-1  
 puna command (exec) 23-1  
 punch driver 14-5, 23-1  
   getting info 23-2  
   logging in 14-5  
   logging out 23-6, 23-9  
   restarting 23-2  
   starting 14-5, 23-1  
   stopping 23-2  
 punch\_end command (exec) 23-2  
 punch\_restart command (exec) 23-2  
 PUSH TO CLUTCH/TRACTORS/PUSH TO  
   DECLUTCH switch (printer) 11-12,  
   22-3

## Q

qedx command (Multics) 27-3  
 quit command (daemon) 22-9, 23-8  
 QUIT command (DPU) 34-3, 34-6  
 quit command (initializer) 22-3, 22-11

## R

RCP 3-11, 19-1, 20-1, 31-1, B-14  
   message 5-5, 7-1, 7-2  
 rcp command (initializer) 19-7, 20-6, 21-8  
 read\_cards command (daemon) 23-7, 23-8  
 reader driver  
   logging in 23-6  
   logging out 23-7, 23-8  
   starting 23-7  
   stopping 23-8  
 reading cards 23-3  
 READY button (disk drive) 38-3  
 READY button (tape drive) 11-1, 11-3, 11-12,  
   19-4  
 ready command (daemon) 24-7

READY light (console) 11-3  
 READY light (disk drive) 20-2, 20-4, 21-4  
 ready message 5-2, 6-1  
 ready mode (DPU) 34-1, B-13  
 readying a disk drive 20-4, 21-4  
 readying a tape drive 19-4  
 rebooting Multics 37-2  
 recognizing a crash  
   in automatic mode 35-1  
   in manual mode 35-2  
 recognizing a system hang 36-1  
 recognizing a system loop 36-1  
 reconfiguration  
   dynamic 3-9, 32-1, B-7  
 reconfigure command (initializer) 19-6, 19-7,  
   19-8, 20-5, 20-6, 21-6, 21-7, 32-2,  
   32-3, 32-4, 32-4.1, 32-5, 32-6, 32-7  
 recording a failure 37-3  
 recovering the system 37-1  
 recovery 3-13, B-13  
 reinit command (daemon) 22-12  
 reinitializing a printer 22-12  
 reinitializing MPC firmware 38-1, 38-2  
 release command (daemon) 23-8  
 releasing an absentee job 30-6  
 reloading 3-13, B-14  
 reloading a FNP 39-2  
 reloading MPC firmware 37-2, 38-1  
 remote device 24-1, B-14  
   logging in 24-7  
   logging in 24-9  
 remote driver 24-1, B-14  
   problems 24-13  
 remote driver process 24-1, B-14  
 remote job entry 23-3  
   see RJE  
 remote stations 25-1, B-14  
   starting up 25-1  
     Type I 24-7  
     Type II 24-9  
     Type I 24-7  
     Type II 24-9  
 remove command (initializer) 6-4, 29-3  
 removing a channel 29-3  
 reprint command (exec) 22-5  
 reprinting a segment 22-5  
 reprocessing a printer request 22-10, 22-11

REQUEST key (console) 5-2, 5-4, 5-8  
 request type 14-1, 14-4, 14-5, 22-13, 24-2,  
     24-6, 24-11, 24-14, 25-2, B-14  
     getting info 22-7, 24-13  
 RESET & BRANCH button (MPC) 38-3  
 RESET button (tape drive) 19-4  
 reset\_tabs command (exec) 6-5  
 resetting time of shutdown 17-6  
 resource control package  
     see RCP  
 responding to a crash  
     in automatic mode 35-1  
     in manual mode 35-2  
 responding to disk problems 38-3  
 responding to IOM alarm 36-11  
 responding to logged-out user 29-3  
 responding to MPC failure 38-1  
 responsibilities  
     operator 4-1  
     system administrator 1-3  
     system maintainer 1-4  
 restart command (daemon) 22-3, 22-10  
 restarting an absentee queue 30-3  
 restarting punch driver 23-2  
 retrieval daemons  
     logging in 27-1, 27-2  
     logging out 27-5  
 retrieve command (backup) 27-4  
 retrieve\_from\_volume command (backup) 27-1,  
     27-5  
 retrieving 3-13, B-14  
 RETURN key (console) 5-2, 5-4, 5-8, 11-4,  
     11-7, 11-10, 11-13  
 RETURN key (DPU) 34-1  
 returning to BCE 13-4, 35-2, 36-1, 36-3, 36-5,  
     36-7, 36-10, 36-12  
 REWIND button (tape drive) 11-3, 11-12, 19-4  
 rewinding a tape 19-5, 19-8  
 ring 1 3-11, B-14  
 ring 4 3-11, B-14  
 rings 3-11, B-14  
     1 3-11, B-14  
     4 3-11, B-14  
     administrative 3-11, B-1  
     user 3-11, B-18  
 RJE 24-1, B-14  
 RJE station 24-1, B-14  
 rload command (MCA) 32-4  
 RLV 3-5, 11-3, 13-3, B-15

root 3-5, B-14  
 root logical volume  
     see RLV  
 root physical volume"  
     see RPV  
 RPV 3-5, 11-3, 37-3, B-15  
 rpv command (BCE) 16-1  
 rtb exec\_com 3-12, 35-2, 36-2, 36-3, 36-5,  
     36-7, 36-10  
 run 2-1

## S

salvager errors 33-1  
 salvager message 5-5, 7-1, 7-3  
 salvagers 3-14, B-15  
     directory 3-14, B-6  
     volume 3-14, B-18  
     volume scavenger 3-14, B-19  
 salvaging 37-3  
 sample\_hs command (daemon) 14-2, 16-1, 22-4,  
     22-6  
 scheduled shutdown 17-1  
 SCU 2-1, 37-1, B-16  
     adding 29-1, 32-2  
     bootload 3-5, 9-1, 32-4, B-3  
     changing state  
         in BCE 12-2  
     deleting 29-1, 32-4  
     setting switches 36-6, 36-8, 36-9, 36-11  
 SCU buttons  
     CABINET SHUTDOWN 8-1  
     INITIALIZE 11-13, 32-3  
     MEM CLEAR 32-3  
     PANEL CLEAR 11-13, 32-3  
     POWER OFF 18-1  
     POWER ON 8-1  
     SYNDROME RESET 32-3  
 SCU PORT NUMBER switch (IOM) 11-2, A-6  
 SCU switches  
     ALARM A-2  
     INTERLACE A-2  
     LWR STORE A-2  
     LWR STORE SIZE A-2  
     MARGIN CONTROL MODE A-2  
     MASK A MASK/PORT ASSIGNMENT 9-1,  
         32-3, 36-6, 36-8, 36-9, 36-11  
     MASK/PORT ASSIGNMENT A-2  
     MEMORY SELECT 32-3  
     MODE 32-3, 36-6, 36-8, 36-9, 36-11, A-2  
     PORT ENABLE 9-1, 32-2, 36-6, 36-8, 36-9,  
         36-11, A-2  
     STORE A A-2  
     STORE A1 A-2  
     STORE B A-2  
     STORE B1 A-2  
 segment 3-1, 3-16, B-15  
 sending message to all users 28-2

sending message to all users at login 28-1  
 sending message to one user 28-1  
 sentinel 7-3, B-15  
 service session 3-5, B-15  
 session 3-5, B-15  
 set\_drive\_usage command (initializer) 20-5, 21-5, 21-6  
 set\_system\_console command (Multics) 5-8, 32-8, 32-10  
 setting absentee slots 30-2  
 setting attended mode 31-2  
 setting automatic mode 31-2  
 setting bootload CPU 9-2  
 setting bootload IOM 9-1  
 setting bootload SCU 9-1  
 setting clock 11-9  
     problems 11-14  
 setting highest numbered absentee queue 30-4  
 setting manual mode 31-3  
 setting maximum number of users 29-1  
 setting SCU switches 36-6, 36-8, 36-9, 36-11  
 setting switches 9-1, A-1  
 setting time 11-10  
 setting unattended mode 31-1  
 shutdown command (daemon) 40-1  
 shutdown command (initializer) 17-4  
 shutdown failure 17-5  
 shutdown\_mpx command (initializer) 39-2  
 shutdowns  
     crash 3-8, 13-4, 35-1, 35-2, B-5, B-16.1  
     nonscheduled 17-5, 28-2  
     orderly 3-8, B-12.1  
     scheduled 17-1  
 shutting down a FNP 39-2  
 shutting down BCE 36-8, 36-11  
 shutting down Data Management system 40-1  
 shutting down immediately 17-5  
 shutting the system down 13-4, 17-1  
 sign\_off command (initializer) 4-4  
 sign\_on command (initializer) 4-3, 16-1  
 signing off 4-4  
 signing on 4-3  
 SKIP button (printer) 11-12, 22-3  
 software 2-1, B-15  
     online 3-9, B-12.1  
     system 3-9, B-17  
 source id 3-13, B-15  
 SOURCE switch (IOM) 9-2, 11-2, 11-13, A-5  
 special characters 4-6  
 ST CU command (DMP/VIP) 34-7, 36-5, 36-10  
 ST CU command (DPU) 34-7, 36-5, 36-10  
 standard time 11-9  
 STANDBY button (tape drive) 11-3, 11-12, 18-1, 19-4  
 STANDBY light (tape drive) 19-2, 19-8  
 START button (disk drive) 20-2, 20-4, 21-2, 21-3, 21-4  
 START button (MPC) 8-1, 11-15  
 START button (printer) 11-12, 22-3  
 START button (tape drive) 8-1, 11-1, 11-3, 11-12, 19-4  
 start command (daemon) 22-13  
 START/STOP switch (disk drive) 8-1, 18-1  
 starting absentee facility 30-1  
 starting coordinator 14-3  
 starting dumper daemons 15-1  
 starting I/O daemons 14-1  
 starting printer driver 14-4, 22-2  
 starting punch driver 14-5, 23-1  
 starting reader driver 23-7  
 starting remote printer 24-2, 24-4  
 starting up a remote station 25-1  
     Type I 24-7  
     Type II 24-9  
 starting up the system 3-8  
 startup command (initializer) 13-1, 13-3  
 startup failure 13-4  
 station command (daemon) 25-2  
 stations  
     remote 25-1, B-14  
         Type I 24-7  
         Type II 24-9  
     RJE 24-1, B-14  
 status command (daemon) 22-6  
 STOP button (disk drive) 18-1  
 STOP button (tape drive) 11-3, 11-12, 19-4  
 STOP CONDITIONS switch (IOM) 9-2  
 stopping a printer 17-3  
 stopping absentee facility 30-1  
 stopping an absentee queue 30-2  
 stopping punch driver 23-2  
 stopping reader driver 23-8

storage device 2-5, B-15  
 storage director B-15  
 storage system 3-1, 3-16, B-15  
 storage system disks 2-5, B-16  
     canceling a mount 21-4  
 storage system hierarchy 3-1, B-16  
 STORE A switch (SCU) A-2  
 STORE A1 switch (SCU) A-2  
 STORE B switch (SCU) A-2  
 STORE B1 switch (SCU) A-2  
 STORE CONTROL switch (IOM) 9-2  
 store unit 2-1, B-16  
     clearing 32-2  
 substty command (initializer) 5-6, 6-4  
 SUBSYSTEM POWER OFF button (disk drive)  
     18-1  
 SUBSYSTEM POWER ON button (disk drive)  
     8-1  
 subvolumes 3-3, 20-1, 21-1  
 supervisor 3-9, B-16  
 SUSP command (DPU) 34-6  
 suspending an absentee job 30-5  
 swapping ttys 6-4, B-16  
 switches 9-1, A-1  
     setting 9-1, A-1  
 SYNDROME RESET button (SCU) 32-3  
 syserr log 3-15, B-16  
 syserr message 3-15, 7-1, 7-2, B-16  
 system administrator 1-3, B-16  
 system administrator's responsibilities 1-3  
 system console 5-1, B-16  
 system control 3-10, B-16  
 system controller 2-1, B-16  
 system controller unit  
     see SCU  
 system crash 3-8, 35-1, 35-2, B-16.1  
 system hang 36-1  
 SYSTEM INIT button (IOM) 11-4  
 SYSTEM INITIALIZE button (IOM) 11-4  
 system log book 3-15, 32-2, 33-1, B-16.1  
 system loop 36-1, B-16.1  
 system maintainer 1-3, B-17  
 system maintainer's responsibilities 1-4  
 system message 7-1  
 system shutdown 3-8, 17-1, B-17  
     crash 3-8, 13-4, 35-1, 35-2, B-5, B-16.1  
     nonscheduled 17-5, 28-2  
     orderly 3-8, B-12.1  
     scheduled 17-1  
 system software 3-9, B-17  
 system startup 3-8, B-17  
 system tape  
     BCE/Multics 3-6, 3-8, 11-1, B-2  
 system\_start\_up.ec 3-12, 14-1, 15-1, 26-4, B-17

T

T & D 1-3, 9-2, 19-8, B-17  
 tag 34-1, B-17  
 tailsheet 22-1, B-17  
 taking a CPU out of step  
     DPS 8 36-5, 36-6, 36-9, 36-10, 36-11  
     L68 36-4, 36-7, 36-8  
 taking a FNP out of service 39-1  
 tape controller 2-5.1, 11-1, 11-7, 11-11, 36-12,  
     37-2, B-17  
 tape drive buttons  
     LOAD 11-1, 19-4  
     LOAD/REWIND 19-4  
     POWER OFF 18-1  
     POWER ON 8-1  
     READY 11-1, 11-3, 11-12, 19-4  
     RESET 19-4  
     REWIND 11-3, 11-12, 19-4  
     STANDBY 11-3, 11-12, 18-1, 19-4  
     START 8-1, 11-1, 11-3, 11-12, 19-4  
     STOP 11-3, 11-12, 19-4  
 tape drives 2-5.1, B-17  
     500 8-1, 11-1, 11-2, 18-1, 19-3  
     600 11-1, 11-2, 19-3  
     610 8-1, 11-1, 11-2, 18-1, 19-3  
     630 11-1, 11-2, 19-3  
     8200 8-1, 11-1, 18-1, 19-4  
     adding 19-6  
     deleting 19-7  
     readying 19-4  
 tape handler 2-5.1, B-17  
 tape reel 2-5.1, 30-5, B-17  
 tapes 2-5.1, B-17  
     authenticating 19-3  
     backup 26-1, B-2  
     BCE/Multics system 3-6, 3-8, 11-1, B-2  
     catchup 26-1, B-3  
     complete dump 26-1, B-4  
     consolidated 26-1, B-5  
     demonstrating 19-5  
     denying 19-3  
     foreign 19-1  
     getting info 19-7  
     hierarchy dump 33-1  
     incremental 26-1, 37-2, B-9  
     mounting 19-1  
     preloading 19-5, 31-1  
     rewinding 19-5  
     wakeup 26-1, B-19

terminals 2-5, B-17  
  crt 2-5, B-5  
  hardcopy 2-5, B-8  
  initializer 3-11, 6-1, B-10  
  message coordinator 6-1, B-12.1  
  printing 2-5, B-13  
  video 2-5, B-18

test and diagnostics  
  see T & D

TEST CONTROL switch (CPU) 9-3

time zone 11-9, 11-14

TM command (DPU) 34-3, 34-7

TM mode (DPU) 34-2, B-18

transparent mode  
  see TM mode

TRBL light (MPC) 38-1

TROUBLE light (CPU) 11-13, 36-12

TROUBLE light (IOM) 11-13, 36-12

TROUBLE light (SCU) 11-13, 36-12

Type I remote station 24-7

Type II remote station 24-9

## U

unattended command (exec) 31-1

unattended mode 19-5, 31-1, 35-1, B-18  
  setting 31-2

unattended service 31-1, B-18

unit 2-1, B-18

UNIT EMERGENCY button (disk drive) 8-1

unit record controller 2-6, 11-11, 37-2, B-18

unit record device 2-5.2, B-18

unit record processor  
  see URP

unload command (initializer) 19-5

unlock mode 5-2, B-18

URP 2-6, B-18

user control 3-12, B-18

user I/O disks 2-5, B-18  
  authenticating 20-3  
  denying 20-4  
  mounting 20-1

user message 6-2

user ring 3-11, B-18

User\_id 3-13, 4-2, B-18

## V

vacate\_pdir\_volume command (initializer) 21-6

vcomp command (exec) 26-7

vcons command (exec) 26-5

video terminal 2-5, B-18

vinc command (exec) 15-2, 16-1, 26-3

VIP command (DPU) 34-3, 34-6

VIP mode (DMP/VIP) 34-5, B-18

VIP mode (DPU) 34-2, 34-5, 36-4, 36-9, B-18

volume backup 3-13, B-18

volume dumping  
  complete 26-7  
  consolidated 26-5  
  incremental 15-2, 15-3, 17-3, 26-3

volume retrievals 27-1

volume salvager 3-14, B-18

volume scavenger 3-14, B-19

volumes

  logical 2-5, 3-1, 13-3, B-11  
  private 3-3, B-13  
  public 3-3, B-13  
  root 3-5, 11-3, 13-3, B-15  
  physical 2-5, 3-1, B-13  
  root 3-5, 11-3, 37-3, B-15  
  sub 3-3, 20-1, 21-1

## W

wakeup dump 26-1, B-19

wakeup hierarchy dumping  
  see incremental hierarchy dumping

wakeup tape 26-1, B-19

wakeup volume dumping  
  see incremental volume dumping

wakeup\_dump command (backup) 17-3

wakeup\_volume\_dump command (backup) 17-3

warm boot 3-5, B-19

warm bootload  
  see warm boot

warn command (initializer) 19-8, 28-1, 28-2,  
  39-1, 39-3

who command (initializer) 14-8, 15-3, 17-2,  
  29-1

word command (initializer) 28-1

workstation 24-1, B-19

## X

x commands  
  see exec commands

## Z

ZERO BASE S.C. PORT NO. switch (IOM)  
  11-2

ZERO BASE S.C. PORT NO. switch (IOM)  
  A-6



HONEYWELL BULL  
Technical Publications Remarks Form

TITLE

MULTICS  
OPERATOR'S GUIDE  
ADDENDUM B

ORDER NO.

GB61-01B

DATED

DECEMBER 1987

ERRORS IN PUBLICATION

[Empty box for errors in publication]

SUGGESTIONS FOR IMPROVEMENT TO PUBLICATION

[Empty box for suggestions for improvement to publication]



Your comments will be investigated by appropriate technical personnel and action will be taken as required. Receipt of all forms will be acknowledged; however, if you require a detailed reply, check here.

**PLEASE FILL IN COMPLETE  
ADDRESS BELOW.**

FROM: NAME \_\_\_\_\_

DATE \_\_\_\_\_

TITLE \_\_\_\_\_

COMPANY \_\_\_\_\_

ADDRESS \_\_\_\_\_

\_\_\_\_\_

CUT ALONG LINE

--- CUT ALONG LINE ---

PLEASE FOLD AND TAPE--  
NOTE: U.S. Postal Service will not deliver stapled forms

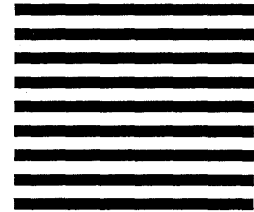


NO POSTAGE  
NECESSARY  
IF MAILED  
IN THE  
UNITED STATES

**BUSINESS REPLY MAIL**  
FIRST CLASS PERMIT NO. 39531 WALTHAM, MA

POSTAGE WILL BE PAID BY ADDRESSEE

**Honeywell Bull Inc.**  
200 Smith Street MS486  
P.O. Box 9199  
Waltham, Massachusetts, 02254-9832



**Honeywell Bull**

# Honeywell Bull

**Corporate Headquarters:**

3800 West 80th St., Minneapolis, MN 55431

**U.S.A.:** 200 Smith Street, MS 486, Waltham, MA 02154

**Mexico:** Hamburgo No. 64, Col. Juarez Delegacion Cuauhtemoc, 06600 Mexico, D.F.

**U.K.:** Great West Rd., Brentford, Middlesex TW8 9DH, England **Italy:** 32 Via Pirelli, 20124 Milano

**Canada:** 155 Gordon Baker Road, North York, Ontario M2H 3P9 **New Zealand:** 14/16 Liverpool Street, Auckland 1

**Asia:** 4/F, Shui on Centre, 6-8 Harbour Rd., Wanchai, Hong Kong **Australia:** 124 Walker Street, North Sydney, N.S.W. 2060