

VAX-11/RSX-11M User's Guide

Order No. AA-D037C-TE

May 1982

This document provides the information needed to use the VAX/VMS MCR command language, execute MCR indirect command files, and use RSX-11M Version 3.2 components under VAX/VMS.

REVISION/UPDATE INFORMATION: This revised document supersedes the VAX-11/RSX-11M User's Guide (Order No. AA-D037B-TE).

SOFTWARE VERSION: VAX/VMS Version 3.0

First Printing, August 1978
Revised, March 1980
Revised, May 1982

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment Corporation assumes no responsibility for any errors that may appear in this document.

The software described in this document is furnished under a license and may be used or copied only in accordance with the terms of such license.

No responsibility is assumed for the use or reliability of software on equipment that is not supplied by Digital Equipment Corporation or its affiliated companies.

Copyright © 1978, 1980, 1982 by Digital Equipment Corporation
All Rights Reserved.

Printed in U.S.A.

The postpaid READER'S COMMENTS form on the last page of this document requests the user's critical evaluation to assist in preparing future documentation.

The following are trademarks of Digital Equipment Corporation:

DEC	DIBOL	RSX
DEC/CMS	EduSystem	UNIBUS
DECnet	IAS	VAX
DECsystem-10	MASSBUS	VMS
DECSYSTEM-20	PDP	VT
DECUS	PDT	digital
DECwriter	RSTS	

ZK2137

HOW TO ORDER ADDITIONAL DOCUMENTATION

In Continental USA and Puerto Rico call 800-258-1710
In New Hampshire, Alaska, and Hawaii call 603-884-6660
In Canada call 613-234-7726 (Ottawa-Hull)
800-267-6146 (all other Canadian)

DIRECT MAIL ORDERS (USA & PUERTO RICO)*

Digital Equipment Corporation
P.O. Box CS2008
Nashua, New Hampshire 03061

*Any prepaid order from Puerto Rico must be placed
with the local Digital subsidiary (809-754-7575)

DIRECT MAIL ORDERS (CANADA)

Digital Equipment of Canada Ltd.
940 Belfast Road
Ottawa, Ontario K1G 4C2
Attn: A&SG Business Manager

DIRECT MAIL ORDERS (INTERNATIONAL)

Digital Equipment Corporation
A&SG Business Manager
c/o Digital's local subsidiary or
approved distributor

Internal orders should be placed through the Software Distribution Center (SDC), Digital Equipment Corporation, Northboro, Massachusetts 01532

CONTENTS

	Page
PREFACE	vii
SUMMARY OF TECHNICAL CHANGES	xi
CHAPTER 1	
INTRODUCTION	
1.1 EXECUTABLE IMAGE TYPES	1-1
1.1.1 Requirements for RSX-11M Task Image Execution under VAX/VMS	1-2
1.2 PROGRAM DEVELOPMENT TOOLS	1-2
1.2.1 RSX-11M Utilities and Files Provided with VAX/VMS	1-3
1.2.2 Additional RSX-11M Facilities	1-4
1.2.3 VAX/VMS Utilities Available under VAX/VMS MCR	1-4
1.3 SUMMARY OF COMMANDS	1-4
1.4 FILES-11 AND TRANSPORTABILITY CONSIDERATIONS	1-7
1.5 FCS AND RMS-11 UNDER VAX/VMS	1-8
1.6 SUPPORT OF RSX-11M DIRECTIVES	1-8
CHAPTER 2	
VAX/VMS ENVIRONMENT	
2.1 PROCESS AND IMAGE	2-1
2.1.1 Process and Image Versus Task	2-2
2.1.2 Login Process, Detached Process, or Subprocess	2-2
2.1.3 Process Name	2-2
2.1.4 Process Identification	2-3
2.2 USER AUTHORIZATION FILE	2-3
2.2.1 VAX/VMS UICs and Protection	2-3
2.2.1.1 File Protection	2-4
2.2.1.2 Process Protection	2-4
2.2.2 Privileges	2-5
2.2.3 Process Priority	2-5
2.3 FILE SPECIFICATIONS	2-5
2.3.1 Device Names	2-7
2.3.1.1 Device Mnemonic	2-7
2.3.1.2 Controller Designation and Unit Numbers	2-8
2.3.2 Directory Names	2-8
2.3.3 Version Numbers	2-8
2.3.4 Defaults in File Specification	2-9
2.4 LOGICAL NAMES	2-12
2.4.1 Process Logical Name Table	2-12
2.4.2 Group Logical Name Table	2-13
2.4.3 System Logical Name Table	2-13
2.4.4 System-Defined Logical Names	2-13
2.5 DEVICE MAPPING	2-15
2.5.1 Mapping RSX-11M Pseudodevice Names	2-15
2.5.2 Mapping RSX-11M Physical Device Names	2-16
2.5.3 The LB Pseudodevice Name and Concealed Devices	2-16
2.6 TERMINAL FUNCTIONS	2-17
2.7 SYNTAX RULES	2-19

2.7.1	Continuing Commands on More Than One Line . . .	2-19
2.7.2	Using Comments in Command Lines	2-19
2.7.3	Truncating Command Names and Keywords	2-19
2.7.4	Placing Keywords	2-20
2.7.5	Specifying File Protection	2-20
2.7.6	Entering Dates and Times	2-21
2.7.6.1	Absolute Time	2-21
2.7.6.2	Delta Time	2-23
2.7.6.3	Absolute and Delta Time Combinations	2-24
2.8	ERROR MESSAGES	2-24
CHAPTER 3	USING VAX/VMS MCR	
3.1	LOGGING IN TO VAX/VMS	3-1
3.1.1	Selecting a Command Interpreter	3-2
3.1.2	LOGIN.COMD File	3-2
3.2	PREPARING TO RUN AN IMAGE	3-2
3.2.1	Allocating Devices	3-2
3.2.1.1	Generic Device Names	3-3
3.2.2	Mounting Volumes	3-3
3.2.3	Assigning Logical Names	3-4
3.2.3.1	Using Logical Names for RSX-11M Task Images	3-4
3.2.3.2	Using Logical Names Instead of the RSX-11M MCR Command REA	3-5
3.2.3.3	Using Logical Names Instead of Redirect	3-5
3.2.3.4	Using Logical Names for VAX-11 Native Images	3-5
3.2.4	Installing RSX-11M Task Images	3-5
3.3	RUNNING IMAGES	3-6
3.4	PROGRAM DEVELOPMENT UNDER VAX/VMS MCR	3-7
3.4.1	Using RSX-11M Components	3-7
3.4.2	Using the File Transfer Program	3-9
3.4.3	Using VAX/VMS Components	3-10
3.5	EXECUTING RSX-11M INDIRECT COMMAND FILES	3-10
3.6	BUILDING RSX-11M IMAGES FOR EXECUTION UNDER VAX/VMS	3-11
3.6.1	Task Builder Switches	3-11
3.6.2	Task Builder Options	3-12
3.6.3	Noncontiguous Task Image Files	3-12
3.6.4	Building RSX-11M Task Images on VAX/VMS for Execution on RSX-11M	3-12
CHAPTER 4	MCR COMMANDS	
4.1	ALLOCATE	4-2
4.2	APPEND	4-4
4.3	ASN	4-7
4.4	BYE	4-9
4.5	CANCEL	4-10
4.6	CONTINUE	4-12
4.7	COPY	4-14
4.8	DEALLOCATE	4-18
4.9	DEBUG	4-19
4.10	DELETE	4-20
4.11	DELETE/ENTRY	4-23
4.12	DEPOSIT	4-25
4.13	DIRECTORY	4-28
4.14	DMO	4-36
4.15	EXAMINE	4-38
4.16	EXECUTE PROCEDURE (@)	4-41
4.17	EXIT	4-43
4.18	INITIALIZE	4-45
4.19	LOGIN PROCEDURE	4-52
4.20	LOGOUT	4-54

CONTENTS

Page

4.21	MOUNT	4-55
4.22	ON SEVERITY-LEVEL	4-66
4.23	PRINT	4-68
4.24	PURGE	4-72
4.25	RENAME	4-74
4.26	RUN (IMAGE)	4-76
4.27	RUN (PROCESS)	4-77
4.28	SET	4-87
4.28.1	SET HOST	4-89
4.28.2	SET PASSWORD	4-90
4.28.3	SET /UIC	4-91
4.29	SHOW	4-92
4.30	STOP	4-94
4.31	STOP/ABORT	4-97
4.32	STOP/ENTRY	4-98
4.33	STOP/REQUEUE	4-99
4.34	SUBMIT	4-100
4.35	TIME	4-106
4.36	TYPE	4-107
4.37	UFD	4-109

CHAPTER 5 INDIRECT COMMAND FILES

5.1	REQUESTING INDIRECT COMMAND FILE EXECUTION	5-7
5.2	SUPPORT OF .IFINS, .IFACT, AND .IFLOA	5-7
5.3	SUPPORT OF .ONERR	5-7
5.4	UNSUPPORTED DIRECTIVES	5-7
5.4.1	.XQT and .WAIT Alternative	5-8
5.4.2	.ONERR Alternative	5-8
5.5	SWITCHES	5-8
5.6	SPECIAL SYMBOLS	5-8
5.7	NUMERIC SYMBOLS	5-10
5.8	LABEL TABLE ENTRIES	5-10
5.9	SYS\$INPUT AND SYS\$COMMAND	5-11
5.10	.ASKN DIRECTIVE	5-11
5.11	.ENABLE DOLLARS DIRECTIVE	5-11
5.12	.PAUSE DIRECTIVE	5-12
5.13	.DELAY DIRECTIVE	5-12
5.14	NESTING MCR INDIRECT COMMAND FILES	5-13
5.15	LEXICAL FUNCTIONS	5-13
5.16	OPERATIONAL NOTES	5-14
5.17	RSX-11S SYSTEM GENERATION REQUIREMENTS	5-15

APPENDIX A PRIVILEGES

APPENDIX B RESOURCE QUOTAS AND LIMITS

INDEX

FIGURES

FIGURE 2-1	Format of VAX/VMS UICs	2-4
3-1	RSX-11M Program Development Cycles Using VAX/VMS MCR	3-7
4-1	Sample of Output of DIRECTORY Command	4-34

TABLES

TABLE	1-1	RSX-11M Utilities Available under DCL and MCR . . .	1-3
	1-2	Summary of MCR Commands	1-5
	1-3	DCL Commands not in MCR	1-7
	2-1	VAX/VMS Device Types	2-7
	2-2	File Specification Defaults	2-9
	2-3	Default File Types	2-9
	2-4	Mapping of RSX-11M Pseudodevice Names	2-15
	2-5	Terminal Function Keys	2-17
	3-1	Valid RSX-11M Version Task Builder Switches . . .	3-12
	4-1	Set Command Options	4-87
	4-2	SHOW Command Options	4-92
	5-1	Summary of VAX/VMS Indirect Command File Processor Directives	5-2
	5-2	Special Symbols	5-9
	5-3	Summary of Lexical Functions	5-14

PREFACE

MANUAL OBJECTIVES

The VAX-11/RSX-11M User's Guide provides the information needed to use the VAX/VMS MCR command language, execute MCR indirect command files, and use RSX-11M Version 4.0 components (for example, the Task Builder) under VAX/VMS. The information in this document is intended to allow RSX-11M users to operate in a familiar environment.

INTENDED AUDIENCE

This document contains the information that an RSX-11M user needs to run existing RSX-11M task images and native VAX-11 images under VAX/VMS, and to develop RSX-11M task images using the VAX/VMS MCR command language.

This document has two prerequisites: a general understanding of RSX-11M and an understanding of the material presented in the VAX/VMS Summary Description and Glossary.

STRUCTURE OF THIS DOCUMENT

Information in this document is organized as follows:

- Chapter 1 describes the requirements for executing RSX-11M task images under VAX/VMS and lists the RSX-11M components available under VAX/VMS.
- Chapter 2 discusses the environment provided by VAX/VMS and describes command language features and syntax rules.
- Chapter 3 describes how to log in to the system, select a command interpreter, prepare task images for execution, run user images (both RSX-11M and VAX-11), and use RSX-11M components for RSX-11M program development.
- Chapter 4 describes the VAX/VMS MCR commands in detail.
- Chapter 5 describes the VAX/VMS MCR directives and provides information about RSX-11S system generation using VAX/VMS as a host system.
- Appendix A describes user privileges, and Appendix B lists resource limits and quotas.

ASSOCIATED DOCUMENTS

The VAX-11 Information Directory and Index provides a complete list of all VAX-11 documents and includes a brief description of each document. For general background about the VAX/VMS system, see the VAX/VMS Primer and the VAX/VMS Summary Description and Glossary.

The following documents may also be useful:

- VAX-11 Information Directory and Index
- VAX-11/RSX-11M Programmer's Reference Manual
- VAX/VMS Primer
- VAX/VMS Command Language User's Guide
- VAX/VMS Guide to Using Command Procedures
- VAX-11 Utilities Reference Manual
- RSX-11M/M-PLUS Utilities Manual

Of the documents listed above, the VAX/VMS Command Language User's Guide provides the most useful additional information, because the VAX/VMS MCR command language and the primary VAX/VMS command language, DCL, have many common features. The VAX-11/RSX-11M User's Guide describes these features to the extent needed to perform the VAX/VMS equivalent of common RSX-11M functions.

Programmers developing RSX-11M task images under VAX/VMS may need to reference the following documents:

- RSX-11M/M-PLUS Guide to Program Development
- RSX-11M/M-PLUS Task Builder Manual
- IAS/RSX-11 ODT Reference Manual
- PDP-11 MACRO-11 Language Reference Manual
- EDT Editor Manual

If you want to know where to find specific information, refer to the RSX-11M/RSX-11S Information Directory and Index or the RSX-11M-PLUS Information Directory and Index. For example, programmers may need additional documents depending on the application and programming language used.

CONVENTIONS USED IN THIS DOCUMENT

In the command and directive descriptions in Chapters 4 and 5, respectively, square brackets ([]) indicate optional syntax. Brackets that are part of directory names, however, do not indicate optional syntax. In addition, certain MCR directives use brackets as part of their required syntax; these cases are noted in the text.

Unless indicated otherwise, all commands described are VAX/VMS MCR commands.

PREFACE

VERSION NUMBERS AND OPERATING SYSTEMS

VAX/VMS compatibility mode allows PDP-11 RSX-11M users and programs to use a VAX-11 system with a minimum amount of difficulty. Compatibility mode is a part of the VAX-11 operating system, VAX/VMS Version 3.0.

Compatibility mode generally supports the functions provided by RSX-11M Version 3.2. Like RSX-11M Version 3.2, compatibility mode does not support the new features added to RSX-11M-PLUS Version 2.0 or RSX-11M Version 4.0.

Except for the new features added since RSX-11M Version 3.2 and certain features that cannot be supported because of inherent differences between the PDP-11 and VAX-11 systems, you can use compatibility mode to provide an environment that resembles RSX-11M on a VAX-11 system.

SUMMARY OF TECHNICAL CHANGES

This manual has been revised to reflect the current support of the RSX-11M operating system under VAX/VMS Version 3.0. Most of these changes refer to commands and their keywords (see Chapter 4).

The following commands have been added to Chapter 4:

RENAME	SET VOLUME
SET DAY	SHOW ERROR
SET DIRECTORY	SHOW MEMORY
SET FILE	SHOW USERS

New keywords with the commands that they modify are listed below as they appear in Chapter 4:

Command	Keyword
INITIALIZE	/LABEL=volume-accessibility
Login Procedure	/[NO]COMMAND[=file-spec]
MOUNT	/[NO]ASSIST /COMMENT="string" /[NO]MESSAGE /[NO]MOUNT_VERIFICATION
PRINT	/[NO]NOTIFY /REMOTE
SET PRINTER	/[NO]PASSALL /[NO]PRINTALL /[NO]WRAP
SET PROCESS	/NAME /RESUME /[NO]SUSPEND
SET QUEUE/ENTRY	/WSEXTENT
SET RMS_DEFAULT	/COMPATIBILITY=keyword /EXTEND QUANTITY /STRUCTURE_LEVEL=n
SET TERMINAL	/[NO]ADVANCED VIDEO /[NO]ALTYPEAHD /[NO]ANSI CRT /[NO]AUTOBAUD /[NO]BLOCK_MODE /[NO]BRDCSTMBX /DEVICE TYPE=terminal-name /[NO]LOCAL ECHO /[NO]MODEM /NODMA /[NO]HANGUP /INQUIRE /XON

SUMMARY OF TECHNICAL CHANGES

SET WORKING_SET	/EXTENT=n /NOADJUST
SHOW DEVICES	/FILES /[NO]SYSTEM
SHOW PROCESS	/CONTINUOUS /IDENT=process-id
SUBMIT	/[NO]KEEP /[NO]LOG FILE[=file-spec] /[NO]NOTIFY /[NO]PRINTER[=queue-name] /WSEXTENT

The SET /UIC command now has new behavior. It still affects the user identification code (UIC) only if you have Change Mode to Kernel (CMKRNL) privilege, but a message now appears if you do not have sufficient privilege to change the UIC default. Your directory default is always changed to the specified directory.

The APPEND, COPY, and TYPE commands each have a new syntax that is consistent with their VAX/VMS DCL counterparts.

MCR supports the VAX/VMS Personal Mail Utility (MAIL).

The following MCR commands used for native mode program development are no longer supported by VAX/VMS MCR:

- LIBRARY - native mode Library Utility
- LINK - native mode Linker
- MACRO - native mode Assembler

The RSX-11M Utility VFY is no longer available under VAX/VMS MCR. To perform the disk verify function, use the VAX/VMS DCL command ANALYZE/DISK.

MCR supports neither the CTRL/T keyboard action nor the SET CONTROL=T command, which have been added to VAX/VMS DCL since VAX/VMS Version 2.0. Use of the ON CONTROL_Y statement is no longer supported.

If your default command language interpreter (CLI) in the user authorization file is MCR and you log in requesting /CLI=DCL, do not use the DCL command SPAWN or the Create Process (\$CREPRC) system service to create a subprocess. These actions would create an MCR subprocess under DCL with erratic results.

The LOGOUT command keywords (qualifiers) /BRIEF and /FULL are no longer supported under MCR. When you log out from an interactive terminal session, the brief format appears. When a batch job logs out, it writes the full format to the log file.

The .OPENR, .PARSE, .READ, and .TESTFILE directives are now supported for indirect command file use.

VAX/VMS compatibility mode software handles version numbers for files in a manner consistent with VAX/VMS. RSX-11M pseudodevice names LB and WK now translate to VAX/VMS system-defined logical names SYS\$\$SYSROOT and SYS\$\$SCRATCH.

For other VAX/VMS compatibility mode components, RSX-11M features that were added since RSX-11M Version 3.2 are generally not supported.

CHAPTER 1

INTRODUCTION

VAX/VMS provides two command languages that you can use to interface with the system:

- DIGITAL Command Language (DCL) -- the DIGITAL standard command language. Previously available only under VAX/VMS, DCL is now available under RSX-11M. VAX/VMS DCL is described in the VAX/VMS Command Language User's Guide. DCL command procedures are discussed in the VAX/VMS Guide to Using Command Procedures.
- Monitor Console Routine (MCR) Command Language -- the command language that provides a command language environment similar to MCR under RSX-11M. VAX/VMS MCR is described in this document, along with its indirect command file processor.

The VAX/VMS MCR command language allows RSX-11M users to make the transition to VAX/VMS using a familiar command language. VAX/VMS MCR and indirect command file capability also permit use of the VAX/VMS system as the host for RSX-11S Version 4.0 system generation.

1.1 EXECUTABLE IMAGE TYPES

VAX/VMS MCR provides the tools needed to develop RSX-11M task images. To develop native mode (VAX-11) images, use DCL. You can use VAX/VMS MCR or DCL to run both RSX-11M and native VAX-11 images. (To run RSX-11M task images under DCL, use the MCR command instead of the RUN command.) Both types of images can also be run from one indirect command file using MCR.

However, for support of MCR indirect command files under VAX/VMS, your command interpreter must be MCR. For information on how to establish MCR as your command interpreter, refer to Section 3.1.1.

VAX/VMS places no restrictions on the native images that you can run using MCR; the system manager imposes any restrictions that exist on a user-by-user basis. The VAX-11 processor and the VAX/VMS operating system impose restrictions on the RSX-11M task images that can execute. The restrictions at both hardware and software levels, which are described in the VAX-11/RSX-11M Programmer's Reference Manual, are summarized below.

INTRODUCTION

1.1.1 Requirements for RSX-11M Task Image Execution under VAX/VMS

The processor requirements for RSX-11M task image execution are as follows:

- The task image must execute in user mode and assume that I-space and D-space are overmapped.
- The task image must not execute any privileged instructions such as HALT or RESET.
- The task image must not issue FIS (Floating Instruction Set) floating-point instructions or EAE (Extended Arithmetic Element) instructions; the software emulates FPP floating-point instructions.

The software requirements are as follows:

- The task image must have been built using the RSX-11M Task Builder to execute in a mapped system. An exception to this rule is a task built as a "multi-user" task. Such tasks cannot be run under RSX-11M (see the VAX-11/RSX-11M Programmer's Reference Manual).
- The task image must not depend on environmental features of RSX-11M that are not available in VAX/VMS, for example, PLAS memory management or significant events.
- The task image must not physically overmap the RSX-11M Executive; the RSX-11M Executive is not present in VAX/VMS.
- The task image must not physically overmap the PDP-11 I/O page; the I/O page is not available in user mode.
- The task image must not use any DECnet-11 functions.
- It must not depend on the 32-word memory granularity of the KT11 memory management kit.

Task images developed under RSX-11M-PLUS, RSX-11D, or IAS that are compatible with RSX-11M Version 3.2 can execute under VAX/VMS if they meet the requirements listed above. However, RSX-11D or IAS task images must be rebuilt using the RSX-11M Task Builder before they can execute under VAX/VMS.

RSX-11M task images do not have to be rebuilt to run under VAX/VMS unless program modification or different task builder options are required. Rebuilding (using the Task Builder) also is required to take advantage of the logical name extensions of FCS and RMS-11 which allow processing of VAX/VMS file specifications.

1.2 PROGRAM DEVELOPMENT TOOLS

VAX/VMS provides the RSX-11M utilities needed for a VAX-11 to function as the host for the development of programs to run on a PDP-11 under RSX-11M, or a VAX-11 under VAX/VMS compatibility mode, or on both systems. Each RSX-11M utility runs in compatibility mode under VAX/VMS. Collectively, these utilities provide the same functions they provide in RSX-11M. In addition, VAX/VMS provides copies of the files and libraries needed for program development.

INTRODUCTION

1.2.1 RSX-11M Utilities and Files Provided with VAX/VMS

VAX/VMS provides the RSX-11M Version 4.0 utilities listed in Table 1-1 as part of the standard system, available under both DCL and MCR.

Table 1-1: RSX-11M Utilities Available under DCL and MCR

RSX-11M Utility	DCL Command	MCR Name
MACRO-11	MACRO/RSX11	MAC
Line Text Editor	*	EDI
Task Builder	LINK/RSX11	TKB
Librarian Utility Program	LIBRARY/RSX11	LBR
Peripheral Interchange Program	*	PIP
File Transfer Utility	*	FLX
ZAP Utility	*	ZAP
File Patch Utility	*	PAT
Disk Save and Compress Utility for Files-11 Structure Level 1	*	DSC1
Bad Block Locator Utility	*	BAD
File Dump Utility	*	DMP
RMS-11 Utilities	*	**

* To run this program from DCL, you can enter the DCL command MCR and then enter the RSX-11M utility name, or enter the command RUN SYS\$SYSTEM:name where the RSX-11M utility name immediately follows the RUN SYS\$SYSTEM: command.

** See the RSX-11M/M-PLUS Utilities Manual.

VAX/VMS also provides the following RSX-11M files and libraries:

- SYSLIB.OLB
- RSXMAC.SML
- ODT.OBJ

With the utilities and files listed above, you can use VAX/VMS as you would use RSX-11M for program development.

For example, you can enter and edit a program using EDT or EDI, assemble it using MACRO-11, build it using the RSX-11M Task Builder, and run it using VAX/VMS as the host system. Task images that are to be debugged using ODT (using the MCR command DEBUG) can be linked with ODT.OBJ.

INTRODUCTION

SYSLIB.OLB contains logical name extensions to FCS and RMS-11 that allow RSX-11M task images to process VAX/VMS file specifications.

RSX-11M task images developed under VAX/VMS can execute under RSX-11M Version 3.2 or 4.0.

1.2.2 Additional RSX-11M Facilities

The following RSX-11M facilities can be performed under VAX/VMS:

- RSX-11S system generation
- Operation of PDP-11 language compilers

Note that in order to obtain the necessary components, a separate license is required for each.

1.2.3 VAX/VMS Utilities Available under VAX/VMS MCR

While operating under the VAX/VMS MCR command language, you can use the VAX/VMS utilities listed below.

- DEBUG - the VAX-11 Symbolic Debugger (if a native mode image was being used).
- DSC2 - the Disk Save and Compress Utility for Files-11 Structure Level 2 volumes.
- EDT - the DIGITAL standard editor, EDT. EDT is a screen-oriented editor.
- MAIL - the VAX/VMS Personal Mail Utility which allows the transfer of messages among users.
- SOS - the line-oriented SOS editor.

For instructions needed to successfully invoke the VAX/VMS utilities, see Section 3.4.3.

1.3 SUMMARY OF COMMANDS

The commands that make up VAX/VMS MCR originate from one of two sources:

- VAX/VMS MCR commands include some of the general-purpose commands found in DCL. Some of these commands are also found in RSX-11M MCR, although command formats may differ slightly (see Chapter 4).
- VAX/VMS MCR commands include some of the commands found in RSX-11M that are not found in DCL. These commands include BYE, TIME, and UFD.

When a command exists only in VAX/VMS MCR or DCL, a command or a utility usually exists in the other command language that performs a similar function. For example, the MCR command ASN and the DCL command ASSIGN perform similar functions, but they have slightly different names and command syntax.

INTRODUCTION

VAX/VMS MCR provides some VAX/VMS DCL features in an RSX-11M command language environment. For an RSX-11M user, VAX/VMS MCR can simplify your transition to VAX-11 operation.

Table 1-2 summarizes the VAX/VMS MCR commands and are grouped according to the functions they provide. In addition to the commands listed, VAX/VMS MCR recognizes commands that invoke the RSX-11M and VAX/VMS components listed in Section 1.2 and the various SET and SHOW commands associated with the VAX-11 system. For example, MAC invokes the MACRO-11 Assembler.

Table 1-2: Summary of MCR Commands

Group	Command	Function
Initialization	ALLOCATE	Reserves an unmounted shareable device or an unallocated nonshareable device for exclusive use by the process
	ASN	Defines or deletes a logical name assignment
	DEALLOCATE	Releases a previously allocated device
	DMO	Releases (dismounts) a volume previously specified in a MOUNT command
	INITIALIZE	Initializes a Files-11 Structure Level 1 or 2 disk volume
	MOUNT	Makes a disk or magnetic tape volume or volume set available for processing
	SET HOST	Establishes a virtual communication link between a terminal and a network node to which the terminal is not directly connected
	SET PASSWORD	Allows users to change their own password
	SET /UIC	Changes the directory default and user identification code (UIC) of the current process
Informational	TIME	Displays the current date and time
Process Control	BYE	Terminates an interactive terminal session
	CANCEL	Cancels scheduled wake-up requests for a process
	CONTINUE	Resumes execution of an interrupted image or an indirect command file
	EXIT	Terminates processing of the current indirect command file

(continued on next page)

INTRODUCTION

Table 1-2 (Cont.): Summary of MCR Commands

Group	Command	Function
	ON	Defines the default course of action to be taken when a command or program in an indirect command file encounters an error condition of a particular severity level
	LOGIN procedure	Verifies your right to use the system and allows you to log in to it
	LOGOUT	Terminates an interactive terminal session
	RUN	Runs images and, optionally, creates detached processes and subprocesses
	STOP	Terminates execution of the image currently running in a process
	SUBMIT	Queues a batch job for processing
Program Development	DEBUG	Invokes ODT for an interrupted RSX-11M task image or the VAX-11 Symbolic Debugger for an interrupted native mode image
	DEPOSIT	Replaces the contents of specified locations in virtual memory
	EXAMINE	Examines the contents of specified locations in virtual memory
File Handling	APPEND	Adds the contents of one or more input files to the end of an output file
	COPY	Creates new files from one or more existing files
	DELETE	Deletes files, entries from print queues, and entries from batch job queues
	DIRECTORY	Displays the names of one or more files in a directory or subdirectory
	PRINT	Queues files for printing
	PURGE	Deletes older versions of a file while saving the specified number of recent versions
	RENAME	Changes the name of a file
	TYPE	Displays the contents of a file at your terminal
	UFD	Creates a directory or subdirectory on a Files-11 Structure Level 1 or 2 disk

INTRODUCTION

Under MCR, only use the VAX/VMS MCR commands found in Table 1-2 or Chapter 4, the VAX/VMS or RSX-11M utilities (see Section 1.2), or user-supplied images. MCR examines command line entries for the appropriate MCR command and, if an MCR command was not entered, assumes an image was requested. MCR attempts to run the image from library SYS\$SYSTEM. Because most RSX-11M utilities need to immediately process the entered command line, MCR passes the image the entire command line, which conflicts with DCL command conventions where DCL passes the image parts of the command line as the image requests them.

In addition to providing an RSX-11M environment for the use of RSX-11M utilities, MCR must allow the execution of user-supplied images. To minimize restrictions for naming user images, MCR assumes images in library SYS\$SYSTEM are valid and does not check whether or not an invalid command was entered.

Because MCR chooses not to check whether an invalid command was entered and must use command line parsing conventions that differ from DCL, if you specify a DCL command that is not found in MCR, MCR attempts to execute the command with erratic results. Problems may not occur immediately after the command is entered and are generally unpredictable. A list of DCL commands not supported by MCR is provided in Table 1-3. If you need to use any of the commands listed in Table 1-3, use DCL instead of MCR.

Table 1-3: DCL Commands not in MCR

= (assignment)	DIFFERENCES	PASSWORD
ACCOUNTING	DISMOUNT (DMO in MCR)	PATCH
ANALYZE*	DUMP	PHONE
ASSIGN* (ASN in MCR)	EDIT/FDL	PLI
ATTACH	EOD	READ
BACKUP	EOJ	REPLY
BASIC*	FORTRAN	REQUEST
BLISS	IF	RUNOFF
CC	INITIALIZE/QUEUE	SEARCH
CLOSE	INQUIRE	SET COMMAND
COBOL*	JOB	SET CONTROL=T
CONVERT*	LIBRARY*	SORT*
CORAL	LINK*	SPAWN
CREATE*	MACRO*	START/QUEUE*
DBO*	MCR	STOP/QUEUE
DDL*	MERGE	SYNCHRONIZE
DEASSIGN* (ASN in MCR)	MESSAGE	UNLOCK
DECK	MONITOR	WAIT
DEFINE	ON CONTROL Y	WRITE
DELETE/QUEUE	OPEN	
DELETE/SYMBOL	PASCAL	

*1. Indicates that the DCL command and all of its subcommands are not provided by MCR. For example, the DCL command ASSIGN and its subcommands ASSIGN/MERGE and ASSIGN/QUEUE are not found in MCR.

1.4 FILES-11 AND TRANSPORTABILITY CONSIDERATIONS

Files-11 is the file structure used for disk volumes under both RSX-11M and VAX/VMS. Files-11 provides two structure levels: Files-11 Structure Level 1 and Files-11 Structure Level 2. RSX-11M supports only Files-11 Structure Level 1; VAX/VMS supports both structure levels.

INTRODUCTION

By default, VAX/VMS creates Files-11 Structure Level 2 volumes; however, it can create and read Files-11 Structure Level 1 volumes (using the INITIALIZE command, /STRUCTURE=keyword). Thus, Files-11 Structure Level 1 volumes can be transported among RSX-11, IAS, and VAX/VMS systems.

The UIC directory format ([g,m]) must be used for RSX-11M transportability (see Section 2.3).

For magnetic tape use, ANSI (17-character) file names are supported under VAX/VMS compatibility mode as described in the VAX/VMS Magnetic Tape User's Guide.

1.5 FCS AND RMS-11 UNDER VAX/VMS

File Control Services (FCS), RMS-11, and RMS-11K (ISAM) are available under VAX/VMS, and all run in compatibility mode. The result is that RSX-11M task images that call FCS/RMS-11 interface with the same record management services that they used in RSX-11M. VAX/VMS converts FCS and RMS-11 QIO requests from the RSX-11M format to VAX/VMS equivalent native mode QIOs. The VAX-11/RSX-11M Programmer's Reference Manual describes the conversion process.

RMS-11 record locking is not supported for RSX-11M task images running in compatibility mode.

1.6 SUPPORT OF RSX-11M DIRECTIVES

The VAX-11/RSX-11M Programmer's Reference Manual describes VAX/VMS support for RSX-11M task images that issue directives to the RSX-11M Executive. Most directives are supported by VAX/VMS; however, the following directives are not:

- PLAS directives, because of substantial differences in hardware memory management techniques.
- GET SENSE SWITCH, because VAX-11 does not have sense switches.
- CONNECT TO INTERRUPT VECTOR, because I/O drivers do not run in compatibility mode under VAX/VMS.
- CONNECT, because VAX/VMS inherently has more process protection than RSX-11M.
- Directives added since RSX-11M Version 3.2, including all directives specific to RSX-11M-PLUS.

CHAPTER 2

VAX/VMS ENVIRONMENT

When using VAX/VMS MCR, you will be aware that the environment provided by this operating system differs from that provided by RSX-11M. For example, you have access to features not available in RSX-11M, and device names are different. This chapter describes some basic VAX/VMS concepts, operational facts, and syntax rules of which you should be aware.

2.1 PROCESS AND IMAGE

When you log in to VAX/VMS, the system automatically creates a process for you. That process provides the virtual address space and control information needed to perform all normal interactive user operations. That is, the process provides the environment within which the user operates under VAX/VMS. Within your process, you run the "images" (programs) needed to perform the desired functions. Because a process can exist without an image, VAX/VMS schedules processes instead of images for execution. A user's privileges, protection, and other information are associated with the process, and thus contribute to the environment (context) within which an image executes.

Under VAX/VMS, an image is a program that has been made executable as a result of linking by either the VAX-11 Linker or the RSX-11M Task Builder (TKB). The VAX-11 Linker is an example of a native image that executes in native mode; the Peripheral Interchange Program (PIP) and TKB are examples of RSX-11M task images that execute in compatibility mode. You can run both native and RSX-11M task images within your process using MCR.

Images in VAX/VMS, like tasks in RSX-11M, reside in files on disk. When you request the execution of an image (for example, by typing the RUN command), VAX/VMS activates that image in your process. As each subsequent image is requested, it overlays the previous image in the process's virtual address space. The VAX/VMS Summary Description and Glossary further describes the concepts of process and image. The sections that follow describe the features of a process that are significant for an RSX-11M user.

2.1.1 Process and Image Versus Task

A process executing an image under VAX/VMS is quite similar in concept to an executing RSX-11M task. However, the following differences exist:

- A process is permanently associated with its user; it is not bound to a particular image.
- A process can serially execute any number of images.
- A process normally remains on the system until the user logs out.

2.1.2 Login Process, Detached Process, or Subprocess

VAX/VMS recognizes three types of processes:

- A login process, which is the process created for you when you log in to the system
- A detached process, which is a fully independent process
- A subprocess, which is a process that you create to run a specified image, but which remains under control of your login process

Normally, you run images in your login process. Because that process is associated with you and your terminal, you can request only one image at a time. In most program development sessions, in which the output from one step (for example, MACRO-11) is input to the next image (for example, the Task Builder), this serial approach is convenient. However, in an application in which images must run concurrently to coordinate their activities, each image must run in a separate process. Therefore, the MCR RUN command allows you to create subprocesses and detached processes, in addition to running images in your current process.

2.1.3 Process Name

Every process has a process name that is created as a result of one of the following events:

- When you log in, VAX/VMS defines a process name that is your user name (for example, FELIX).
- When a native image running in your process issues a system service request to change the process name.
- When you run an RSX-11M task image, VAX/VMS creates a process name if the image has a task name whose first 3 characters do not form an ellipsis (...) in its image label block. Utilities do not cause the process name to change because these names start with an ellipsis (for example, ...PIP).

Normally, a process name remains for the duration of the process or until a new name is specified. When you run an RSX-11M task image that has a task name that does not start with an ellipsis, the task name becomes the process name for the duration of that image. When the image terminates, VAX/VMS restores the previous name of the process.

VAX/VMS ENVIRONMENT

VAX/VMS qualifies a process name by the group number of its user identification code (UIC). Therefore, process names need be unique only within the group.

2.1.4 Process Identification

VAX/VMS assigns every process and subprocess a process identification number that is unique throughout the system. The identification of a process does not change as long as the process exists.

You must specify a process identification to affect processes outside your group.

You can use the SHOW PROCESS command to display your process name and identification. You can use the SHOW SYSTEM command, or the SHOW PROCESS command with the /IDENT keyword, to display the name and identification of other processes.

2.2 USER AUTHORIZATION FILE

The VAX/VMS system manager maintains a file called the user authorization file (UAF). It contains one entry for each user who is allowed access to the system. Each entry includes the following information:

- User name
- User password
- User identification code (UIC)
- User account name
- User default disk device
- User default command language interpreter (CLI)
- User's default directory name
- Name of an indirect command file or command procedure to execute at login, for example, LOGIN.CMD if your default command language is MCR
- Default file protection to be applied to newly created files
- User privileges allowed
- Process base priority
- Amount of resources available to user

When you log in, VAX/VMS uses your user authorization file entry to associate your UIC, privileges, and priority with the process it creates for you.

2.2.1 VAX/VMS UICs and Protection

Under VAX/VMS, a UIC is a unique 32-bit numeric identification of a particular user. It consists of a 16-bit octal group number and a 16-bit octal member number, as illustrated in Figure 2-1.

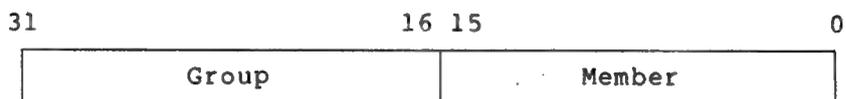


Figure 2-1: Format of VAX/VMS UICs

UIC group and member numbers range from 0 to 377 (octal).

VAX/VMS UICs are used to determine file and process protection.

2.2.1.1 File Protection - When you create a file, your UIC becomes the file-owner UIC. You specify read, write, execute, and delete file protection for system, owner, group, and world based on your UIC. This type of file protection is similar to that under RSX-11M.

Like protection under RSX-11M, VAX/VMS protection for each user category is represented by the four 4-bit fields RWED. The only difference in the interpretation of VAX/VMS and RSX-11M protection is in the E field. Under VAX/VMS, this field determines whether a user has the right to execute files that contain executable program images (either VAX-11 or RSX-11M images). When applied to an entire volume, the E field determines whether users can create files on that volume. (Under RSX-11M, the E field determines whether or not a file can be extended.)

Under VAX/VMS, your UIC is not equivalent to your default directory name. That directory name is specified as a separate entry in the user authorization file (UAF). Section 2.3.2 discusses directory names in greater detail.

2.2.1.2 Process Protection - VAX/VMS also uses UICs for process protection. Process protection determines which processes you can affect using any of the following means:

- An MCR or DCL command that accepts a process name as a parameter (for example, the commands CANCEL and STOP)
- An RSX-11M task image that issues directives specifying another process as the target (for example, ABORT TASK)
- A native image that issues system services specifying another process as the target (for example, the Suspend Process system service)

A process can always affect the subprocesses that it creates.

In addition, using UIC-based process protection in conjunction with process privileges, VAX/VMS allows a process to affect other processes in the following categories:

- Processes having the same UIC group number as the requesting process
- All processes in the system regardless of UIC

2.2.2 Privileges

RSX-11M recognizes users as privileged or nonprivileged based on the group number of the UIC. Group numbers equal to or less than 10 (octal) are privileged. Once a user has logged in, RSX-11M associates the privileged or nonprivileged status with the terminal from which the user is issuing commands.

VAX/VMS privileges differ from RSX-11M privileges in two respects:

- User privileges are associated with the process VAX/VMS creates for you, rather than being associated with your terminal.
- Users are not categorized as privileged or nonprivileged; rather, the VAX/VMS system manager can associate any of a wide range of privileges with each user, depending on that user's needs.

VAX/VMS controls the functions that you are allowed to request based on the privileges specified in your user authorization file entry. All of the privileges defined by VAX/VMS are listed in Appendix A. The privileges that an MCR user requires are normally the same as those that a DCL user requires.

During execution, if an image attempts to use a function for which the user does not have the privilege, the image receives an error status.

The individual command descriptions in Chapter 4 indicate the privileges required for each command. If you attempt to use a command for which you do not have the appropriate privilege, the system issues an error message. Error messages are described and listed alphabetically in the VAX/VMS System Messages and Recovery Procedures Manual.

Two privileges are GROUP and WORLD process control. A process with GROUP process control privilege can affect other processes that have the same UIC group number. A process with WORLD process control privilege can affect any process in the system regardless of its UIC.

2.2.3 Process Priority

The user authorization file entry also specifies the priority at which your process executes. VAX/VMS priorities range from 1 through 15 for normal processes and from 16 through 31 for real-time processes. In RSX-11M, a task has a priority assigned to it. When a user runs a task, the task executes at its own priority. Under VAX/VMS, however, all images execute at the requesting user's process priority.

2.3 FILE SPECIFICATIONS

RSX-11M and VAX/VMS file specifications are similar. The main differences are in the device and directory fields. Differences in the device field are related to how VAX/VMS uses a letter to designate the controller. Differences in the directory field result from differences in Files-11 Structure Levels 1 and 2. VAX/VMS file specifications have the following format:

```
node::device:[directory]filename.type;version
```

VAX/VMS ENVIRONMENT

- node** Node name. The node name is a 1- to 6-character alphanumeric string that must contain at least one alphabetic character. The node name identifies a location on the network and is separated from the device name by two colons (::). You can optionally include a 3- through 42-character access control string enclosed in quotation marks (") in the following format:
- ```
node"access-control-string"::
```
- The DECnet-VAX User's Guide describes the use of access control strings and also network file specification.
- device** Device name. Device names are specified in the following format:
- ```
ddcu:
```
- dd is a mnemonic for the device type.
c is a controller designation.
u is the device unit.
- The maximum length of the device name field is 15 characters.
- The device name is separated from the directory name by a colon (:).
- directory** Directory name. Directory names are specified in one of the following formats:
- ```
[g,m]
[name]
[name.name]
```
- Under VAX/VMS, all of the above formats can be used for Files-11 Structure Level 1 and 2 disks; however, only the [g,m] format can be used under RSX-11M. The [g,m] format consists of two numbers, separated by a comma, that represent the group number (g) and member name (m) that make up the user identification code (UIC), as described in Section 2.2.1.
- The other two directory formats allow the use of letters and numbers to form a directory name. A period present in directory name indicates that the directory is a subdirectory of a higher-level directory. Each period in a directory name denotes a successively lower subdirectory level; up to seven subdirectory levels are allowed.
- Angle brackets (<>) can be used instead of square brackets ([]) to delimit the directory name. For example, specifying [SMITH] and <SMITH> are the same.
- filename** File name. A file name consists of 0 to 9 alphanumeric characters.
- file type** File type. A file type consists of 0 to 3 alphanumeric characters.
- file version** File version number. Version numbers are decimal numbers ranging from 1 through 32767. You can use either a semicolon (;) or a period (.) to separate a file type from a file version number.

File names, file types, and version numbers apply only to files on mass storage volumes. Directory names apply only to files on disk volumes. For record-oriented devices, only the device name field of the file specification is required.

You can use wild card characters in the file name, file type, and version number fields of a file specification. For further information concerning wild card characters, refer to the VAX/VMS Command Language User's Guide.

For repetitive operations, you can use logical names to specify all or part of a file specification (see Section 2.4).

2.3.1 Device Names

VAX/VMS device names consist of a device mnemonic, a controller designation, and a unit number in the format:

ddcu:

2.3.1.1 Device Mnemonic - The device mnemonic can be any one of those listed in Table 2-1.

Devices supported by both VAX/VMS and RSX-11M have the same mnemonics under both systems.

Table 2-1: VAX/VMS Device Types

| Mnemonic | Device Type                              |
|----------|------------------------------------------|
| CR       | Card Reader                              |
| CS       | Console Storage Device                   |
| DB       | RP05, RP06 Disk                          |
| DD       | TU58 Cartridge Tape                      |
| DL       | RL02 Cartridge Disk                      |
| DM       | RK06, RK07 Cartridge Disk                |
| DQ       | RB02 Cartridge Disk, RB80 Disk           |
| DR       | RM03, RM05, RM80, RP07 Disk              |
| DU       | UDA Disk                                 |
| DX       | RX01 Floppy Diskette                     |
| DY       | RX02 Floppy Diskette                     |
| LA       | LP11-K Laboratory Peripheral Accelerator |
| LC       | Line Printer on DMF32                    |
| LP       | Line Printer on LP11                     |
| MB       | Mailbox                                  |
| MF       | TU78 Magnetic Tape                       |
| MS       | TS11 Magnetic Tape                       |
| MT       | TE16, TU45, TU77 Magnetic Tape           |
| NET      | Network Communications Logical Device    |
| NL       | System "Null" Device                     |
| OP       | Operator's Console                       |
| RT       | Remote Terminal                          |
| TT       | Interactive Terminal on DZ11             |
| TX       | Interactive Terminal on DMF32            |
| XA       | DR11-W General Purpose DMA Interface     |
| XD       | DMP-11 Synchronous Communications Line   |
| XF       | DR32 Interface Adapter                   |
| XG       | DMF32 Synchronous Communication Line     |
| XJ       | DUP11 Synchronous Communications Line    |
| XM       | DMC11 Synchronous Communications Line    |

Usually drivers used for any of the devices listed above are loaded during system generation. However, if a driver is needed but it has not been loaded, ask the system manager for assistance.

**2.3.1.2 Controller Designation and Unit Numbers** - The device controller is designated by an alphabetic letter (A through Z). For example, MTA designates magnetic tape controller A.

Unit numbers are decimal numbers ranging from 0 to 65535. VAX/VMS unit numbers start at 0 for each controller; for example, a system can have two tape units on two different controllers designated as MTA0 and MTB0. Each is unit 0 on its controller.

### 2.3.2 Directory Names

Directory names are represented in the UIC ([g,m]) format or by a 1- to 9-character alphanumeric string. Using subdirectories, you can specify as many as seven directory levels after the main directory name. For example, a second-level directory has the following format:

[name.name]

When subdirectories are used, the comma in a UIC-format directory or subdirectory name is omitted; leading zeros are required. An example of a directory name is [122020.MSG]. Directories are described further in the VAX/VMS Command Language User's Guide.

Use the MCR command UFD as described in Chapter 4 to create directories on a Files-11 Structure Level 2 or Files-11 Structure Level 1 volumes. You may need certain access rights or privileges to create a directory, as described in Chapter 4.

Any volumes that are to be transported between VAX/VMS and RSX-11M systems must have directories in the UIC format; the comma can be omitted. For example, [123050] and [123,050] are equivalent on both systems.

### 2.3.3 Version Numbers

RSX-11M displays version numbers in octal. VAX/VMS displays them in decimal. However, both systems maintain version numbers in binary; version numbers are compatible internally. To determine version numbers when moving files between systems, use PIP on either VAX/VMS or RSX-11M to list file name and version information as it appears in that system. The PIP supplied with VAX/VMS has been modified to print version numbers in decimal. You can also use the DIRECTORY command to display the contents of a directory or subdirectory.

Before you use negative version numbers, make sure you use them according to VAX/VMS conventions. Under RSX-11M, version number -0 and 0 have identical meanings as the newest version of the file. Under VAX/VMS, however, only 0 means the newest version. Under VAX/VMS, version -0 means oldest version number, not -1 as under RSX-11M. You can use -1, -2, and so forth under VAX/VMS to denote successively older versions. Under RSX-11M, -1 means oldest version and all lower negative values are illegal. In summary, you can use 0 as a means of specifying the newest version, -0 to specify the oldest version, -1 to specify the next to the newest version, and so forth.

2.3.4 Defaults in File Specification

When a field of a file specification is omitted, VAX/VMS supplies defaults. Table 2-2 lists the defaults for each field.

Table 2-2: File Specification Defaults

| Field        | Default                                                                                                                                                                                                                                                                                                                                            |
|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| node         | The local system is your default.                                                                                                                                                                                                                                                                                                                  |
| device       | The disk is established as your default by (1) your entry in the user authorization file, or (2) execution of a SET DEFAULT command.                                                                                                                                                                                                               |
| directory    | The directory is established as your default by (1) your entry in the user authorization file or (2) execution of a SET DEFAULT or the SET /UIC command.                                                                                                                                                                                           |
| file name    | No defaults are applied to the first file name in an input file specification. Most commands apply default output file names based on the file name of an input file.                                                                                                                                                                              |
| file type    | File type that is the default for the particular command or component being used. Table 2-3 lists default file types.                                                                                                                                                                                                                              |
| file version | The rules for establishing default file version numbers are the same as those used in RSX-11M.<br><br>For input files, the system assumes the most recent version, that is, the highest version number.<br><br>For output files, the system increases the version number by 1 for existing files and supplies a version number of 1 for new files. |

Table 2-3: Default File Types

| File Type  | File Contents                                                 |
|------------|---------------------------------------------------------------|
| AIJ        | Journal file output by the VAX-11 Data Base Operation Utility |
| ANL        | Output file for the VAX/VMS DCL command ANALYZE               |
| BAS        | Source input for the BASIC-PLUS and VAX-11 BASIC Compiler     |
| BJL        | Back-up journal file                                          |
| B2S        | Source input for the PDP-11 BASIC-PLUS-2/VAX Compiler         |
| B32 or BLI | Source input for the VAX-11 BLISS-32 Compiler                 |
| C          | Source input for the VAX-11 C Compiler                        |

(continued on next page)

VAX/VMS ENVIRONMENT

Table 2-3 (Cont.): Default File Types

| File Type | File Contents                                                                                                                                                                      |
|-----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| CBL       | Source input to the PDP-11 COBOL-74/VAX Compiler                                                                                                                                   |
| CLD       | Command language description file                                                                                                                                                  |
| CMD       | RSX-11M indirect command file to be executed with the @ (Execute Command Procedure) under the MCR command interpreter                                                              |
| COB       | Source input for the VAX-11 COBOL and VAX-11 COBOL-74 Compiler                                                                                                                     |
| COM       | VAX/VMS command procedure file that is executed with the @ (Execute Procedure) command, or submitted for batch execution with the SUBMIT command under the DCL command interpreter |
| COR       | Source input for the VAX-11 CORAL 66 Compiler                                                                                                                                      |
| DAT       | Input or output data file                                                                                                                                                          |
| DBS       | VAX-11 Data Base Operation storage area file                                                                                                                                       |
| DDL       | VAX-11 Data Definition Language file                                                                                                                                               |
| DIC       | VAX-11 DATATRIEVE/VAX dictionary file                                                                                                                                              |
| DIF       | Output listing created by the DIFFERENCES command                                                                                                                                  |
| DIR       | Directory file                                                                                                                                                                     |
| DIS       | Distribution list for the MAIL command                                                                                                                                             |
| DMP       | Output from the DUMP command                                                                                                                                                       |
| EDT       | Initialization command input for the EDT editor                                                                                                                                    |
| EXE       | Image created by the VAX-11 Linker or the VAX/VMS version of the RSX-11M Task Builder                                                                                              |
| FDL       | VAX-11 File Definition Language file                                                                                                                                               |
| FLB       | VAX-11 FMS forms library                                                                                                                                                           |
| FOR       | Source input for the VAX-11 FORTRAN Compiler                                                                                                                                       |
| FTN       | Source language input for the PDP-11 FORTRAN IV or FORTRAN IV-PLUS Compiler                                                                                                        |
| HLB       | Help text library file                                                                                                                                                             |
| HLP       | Input source file for help libraries                                                                                                                                               |
| JNL       | Journal file output from the VAX-11 PATCH Utility                                                                                                                                  |
| JOU       | Journal file/audit trail from EDT                                                                                                                                                  |
| L32       | Precompiled library for VAX-11 BLISS-32                                                                                                                                            |

(continued on next page)

VAX/VMS ENVIRONMENT

Table 2-3 (Cont.): Default File Types

| File Type  | File Contents                                                                             |
|------------|-------------------------------------------------------------------------------------------|
| LIB        | Input file containing VAX-11 COBOL-74 source statements                                   |
| LIS        | Listing file produced by a VAX-11 Compiler or assembler                                   |
| LOG        | Batch job output file                                                                     |
| LST        | Listing file produced by a PDP-11 Compiler or assembler                                   |
| MAC        | Source input to the PDP-11 MACRO-11 Assembler                                             |
| MAI        | Mail message file                                                                         |
| MAP        | Memory allocation (map) listing produced by the VAX-11 Linker or the RSX-11M Task Builder |
| MAR        | Source input to the VAX-11 MACRO Assembler                                                |
| MEM        | Output file for DIGITAL Standard Runoff Utility                                           |
| MLB        | VAX/VMS or RSX-11M macro library                                                          |
| MSG        | Source file that specifies the text of messages                                           |
| OBJ        | Object module produced by a PDP-11 or VAX-11 assembler or Compiler                        |
| ODL        | Overlay description input to RSX-11M Task Builder                                         |
| OLB        | VAX/VMS or RSX-11M object module library                                                  |
| OPT        | Options file for input to the VAX-11 Linker                                               |
| PAR        | A SYSGEN parameter file                                                                   |
| PAS        | Source input for the VAX-11 PASCAL Compiler                                               |
| PLI        | Source input for the VAX-11 PL/I Compiler                                                 |
| RNO        | Source input for DIGITAL Standard Runoff                                                  |
| ROO        | VAX-11 Data Base Operation file                                                           |
| R32 or REQ | VAX-11 BLISS-32 source file required for compilation                                      |
| STB        | Symbol table file created by the VAX-11 Linker or the RSX-11M Task Builder                |
| SYS        | System image                                                                              |
| TEC        | TECO indirect command file                                                                |
| TLB        | Text library                                                                              |

(continued on next page)

Table 2-3 (Cont.): Default File Types

| File Type | File Contents                                                                             |
|-----------|-------------------------------------------------------------------------------------------|
| TMP       | Temporary file                                                                            |
| TMn       | SOS temporary file (n is a digit)                                                         |
| TXT       | Input file for text libraries or output file for the MAIL command                         |
| UPD       | Update file of changes for a VAX-11 MACRO source program; also input to the SUMSLP editor |

## 2.4 LOGICAL NAMES

The VAX/VMS logical name capability is an extension of that available under RSX-11M. Section 3.2.3 describes the use of logical names under VAX/VMS. Using the VAX/VMS MCR command ASN (Assign), you can assign a logical name to the following:

- A physical device name, logical device name, or pseudodevice name
- A file specification or a portion of one

VAX/VMS categorizes logical names differently from RSX-11M. Rather than categorizing them as login, local, and global logical names, VAX/VMS describes them as process-local, available to all processes within a group, and available to all processes in the system. VAX/VMS maintains a logical name table for each category:

- Process logical name table
- Group logical name table
- System logical name table

The SHOW LOGICAL command displays the logical names and their equivalence names from the three tables.

When VAX/VMS encounters a logical name in a command or program, it translates the logical name to its equivalence name. To do so, it searches the process, group, and system tables, in that order, and uses the first match it finds. Thus, entries in the process table take precedence over those in the group and system tables, and entries in the group table take precedence over those in the system table. Once VAX/VMS has translated a logical name, it applies any appropriate file specification defaults.

### 2.4.1 Process Logical Name Table

Each process logical name table contains the logical names that are local to each process. Once an entry is assigned to the process logical name table, it is available to all images that run in the process until it is deassigned, or until the process is deleted (for example, as a result of logging out).

#### 2.4.2 Group Logical Name Table

Each group logical name table contains the logical names that are available to all processes that have the same group number in their UICs as the process that created the entry. Unlike RSX-11M, VAX/VMS does not use the identification of the initiating terminal to determine whether a process has access to the group logical name table. Entries remain in this table until they are explicitly deleted.

The GRPNAM privilege is required for a user to place a name in or delete a name from the group logical name table.

#### 2.4.3 System Logical Name Table

The system logical name table contains entries available to all processes in the system. The manner in which entries in this table are used is similar to the manner in which global assignments are used under RSX-11M. Entries remain in the table until they are explicitly deleted.

The SYSNAM privilege is required for a user to place a name in or delete a name from the system logical name table.

#### 2.4.4 System-Defined Logical Names

VAX/VMS defines logical names at the process and system levels. The system assigns logical names for every process created during login or for execution of a batch job. VAX/VMS automatically defines the following logical names for each process logical name table:

- SYS\$INPUT -- command input stream for the process. For an interactive user, SYS\$INPUT is assigned to the terminal. For a batch job, SYS\$INPUT is assigned to the batch input stream.
- SYS\$OUTPUT -- command output stream for the process. For an interactive user, SYS\$OUTPUT is assigned to the terminal. For a batch job, SYS\$OUTPUT is assigned to the batch job log file. For indirect command files, you can temporarily assign SYS\$OUTPUT to a different file as described by the Execute Procedure in Chapter 4.
- SYS\$COMMAND -- original SYS\$INPUT for a process. When a process is executing an indirect command file, SYS\$INPUT is assigned to that file; SYS\$COMMAND remains assigned to the original command stream.
- SYS\$error -- default output stream to which the system writes messages. For an interactive user, SYS\$error is assigned to the terminal. For a batch job, SYS\$error is assigned to the batch job log file.
- SYS\$DISK -- default disk device established at login or established by the SET DEFAULT command described in the VAX/VMS Command Language User's Guide.

- SYS\$LOGIN -- created at login time to refer to the device and directory of the login process which is specified in each user authorization file.
- SYS\$NET -- defined only during VAX-11 DECnet task-to-task communication, the source process that invokes a target process. When opened by the target process, SYS\$NET represents the logical link over which that process can exchange data with its partner. See the DECnet-VAX User's Guide for additional information.
- SYS\$SCRATCH -- default directory and device to which temporary files are written.
- TT -- default device name for terminals. With interactive terminal use, TT is assigned to the SYS\$COMMAND device. When a process executes an indirect command file, TT is assigned to the "null" device.

VAX/VMS defines the following entries for the system logical name table:

- DBG\$INPUT -- the default input stream for the VAX-11 Symbolic Debugger. Equated to SYS\$INPUT at login.
- DBG\$OUTPUT -- the default output stream for the Symbolic Debugger. Equated to SYS\$OUTPUT at login.
- SYS\$DISK -- the VAX/VMS system disk where SYS\$SYSTEM resides.
- SYS\$ERRORLOG -- the device and directory name of error log data files.
- SYS\$EXAMPLE -- the device and directory name of system examples.
- SYS\$HELP -- the device and directory of system help files.
- SYS\$INSTRUCTION -- the device and directory of the computer-aided instruction (CAI) images.
- SYS\$LIBRARY -- the device and directory that contains system libraries.
- SYS\$MAINTENANCE -- the device and directory name of system maintenance files.
- SYS\$MANAGER -- the device and directory name of system manager files.
- SYS\$MESSAGE -- the device and directory of system message files.
- SYS\$NODE -- the node name for the local system, if DECnet is active on the system.
- SYS\$SHARE -- the device and directory of system shareable images.
- SYS\$SYSDEVICE -- the VAX/VMS system disk containing system directories.

## VAX/VMS ENVIRONMENT

- SYSSYSROOT -- the device and root directory for the system directories.
- SYSSYSTEM -- the device and directory of the system disk that contains the operating system program and procedures.
- SYSTEST -- the device and directory name of User Environment Test Package (UETP) files.
- SYSUPDATE -- the device and directory name of system update files.

### 2.5 DEVICE MAPPING

Device mapping is the technique of equating an RSX-11M device name with a VAX/VMS device name. VAX/VMS automatically performs device mapping for two categories of device names:

- RSX-11M pseudodevice names (for example, TI)
- RSX-11M physical device names (for example, DB2)

Mapping occurs when an image uses an RSX-11M device name, or when a command issued either interactively or from an indirect command file that uses an RSX-11M device name.

#### 2.5.1 Mapping RSX-11M Pseudodevice Names

VAX/VMS maps RSX-11M pseudodevice names to VAX/VMS logical names that serve similar functions. Table 2-4 shows the correspondence between RSX-11M pseudodevice names and VAX/VMS logical names.

Table 2-4: Mapping of RSX-11M Pseudodevice Names

| RSX-11M Pseudodevice Name | VAX/VMS Logical Name                      |
|---------------------------|-------------------------------------------|
| TI (for input)            | SYSSINPUT                                 |
| TI (for output)           | SYSSOUTPUT                                |
| CO                        | SYSSCOMMAND                               |
| CL                        | SYSSERROR                                 |
| SY                        | SYSSDISK                                  |
| LB                        | SYSSSYSROOT                               |
| WK                        | SYSSSCRATCH                               |
| SP                        | Device assigned by VAX/VMS system manager |
| OV                        | Device containing image file              |

Devices TI, CO, and CL have permanently open files associated with them. An RSX-11M image interacts with the process-permanent files by issuing I/O requests on TI, CO, and CL that VAX/VMS automatically converts to requests on SYS\$INPUT, SYS\$OUTPUT, SYS\$COMMAND, and SYS\$ERROR. See Section 2.5.3 for information about the LB pseudodevice name.

### 2.5.2 Mapping RSX-11M Physical Device Names

You can control the association of an RSX-11M physical device name with an actual VAX/VMS device, or VAX/VMS can perform the association automatically. To associate an RSX-11M device name with a specific VAX/VMS device unit, you can issue an MCR ASN (Assign) or MOUNT command that specifies the RSX-11M device name as the logical name for the device, as described in Chapter 3.

If you do not use logical names, VAX/VMS automatically maps the RSX-11M device name to a VAX/VMS device name. VAX/VMS performs the mapping by retaining the device type (for example, MT) and converting the RSX-11M unit number into the corresponding VAX/VMS controller letter and unit number.

VAX/VMS performs the mapping to a physical device by converting the RSX-11M octal unit number to decimal and dividing it by 16 (decimal). The quotient is added to the ASCII value representing the character A. The result is the controller letter. The remainder becomes the VAX/VMS decimal unit number. For example, RSX-11M devices TT0 and DB22 become VAX/VMS devices TTA0 and DBB2, respectively, as follows.

TT0 to TTA0:

```
Convert octal 0 to decimal = 0
Controller and unit = 'A'+(0/16)='A'+0 with a remainder of 0
 'A'+0 = 'A' = controller
 0 = unit number
```

DB22 to DBB2:

```
Convert octal 22 to decimal = 18
Controller and unit = 'A'+(18/16)='A'+1 with a remainder of 2
 'A'+1='B' = controller
 2 = unit number
```

VAX/VMS performs this conversion when assigning an I/O device to an RSX-11M image when no logical name exists.

### 2.5.3 The LB Pseudodevice Name and Concealed Devices

A new type of device has been defined in VAX/VMS Version 3.0. This new device, called a "concealed device," has two underscore characters (\_\_) preceding the physical device name.

Logical name translation is not as tightly bound to actual device names as it was in previous versions of VAX/VMS. The device name parameter provided to a VAX/VMS system service is now fully translated by the system service. In addition, VAX-11 RMS returns, as the device name, a logical name that translates to a concealed device.

A new directory concept was introduced with VAX/VMS Version 3.0 that is associated with concealed devices, called a "root directory." A concealed device can refer to either an actual device or to a device and directory combination, called a "rooted device." Using logical names, a device can be referenced as a rooted device or an actual device, but not both at the same time. When you use a rooted device logical name, such as the system-defined logical name SYSSYSROOT, only subdirectories of the root directory on that device can be accessed, not the Master File Directory (MFD) and all of its associated first-level directories.

Most applications are not affected by the addition of rooted directories. However, the RSX-11M pseudodevice name LB translates to a rooted device (SYSSYSROOT) to allow access to system libraries. Initially, you can only reference subdirectories of the root directory using the LB name, such as [1,1] and [1,2]. If an RSX-11M task image attempts to access a file in another directory using the LB name as part of the file specification, it will unsuccessfully attempt to access that directory as a subdirectory of the root structure and not the MFD, unless the LB logical name assignment is changed.

If an RSX-11M task image uses the LB name, you can use the MCR command ASN to change the LB rooted device logical name to a logical name that translates to an actual device. However, attempts to access system library directories while LB is assigned to an actual device will fail. Simply deassign the LB name to reset it to its initial translation when needed.

For additional information on concealed devices, see the VAX-11 Record Management Services Reference Manual.

## 2.6 TERMINAL FUNCTIONS

Table 2-5 lists the terminal control key sequences that VAX/VMS recognizes. With the exception of CTRL/Y and CTRL/X, these terminal functions are essentially the same as those available under RSX-11M. CTRL/Y always causes a return to the command interpreter. CTRL/X cancels the current line and deletes data in the type-ahead buffer. Under VAX/VMS, you do not have to wait for one command or program to terminate before typing the next request. Unprocessed commands are stored in the type-ahead buffer.

Table 2-5: Terminal Function Keys

| Key                                      | Function                                                                                                                                                                                                                                                                                                                                |
|------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>RETURN</p> <p>Control characters:</p> | <p>(Carriage return) Transmits the current line to the system for processing.</p> <p>Before a terminal session, initiates login sequence.</p> <p>Define functions to be performed. You enter a control character by pressing the CTRL key while you press a letter key. Most CTRL/x key sequences are echoed on the terminal as ^x.</p> |

(continued on next page)

Table 2-5 (Cont.): Terminal Function Keys

| Key    | Function                                                                                                                                                                                                                                                                                                                                             |
|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| CTRL/C | <p>Before a terminal session, initiates a login sequence.</p> <p>During command entry, cancels command processing.</p> <p>Certain system and user programs provide special routines to handle CTRL/C interrupts. If CTRL/C is pressed to interrupt a program that does not handle CTRL/C, CTRL/C has the same effect as CTRL/Y and echoes as ^Y.</p> |
| CTRL/I | <p>Duplicates the function of the TAB key.</p>                                                                                                                                                                                                                                                                                                       |
| CTRL/K | <p>Advances the current line to the next vertical tab stop.</p>                                                                                                                                                                                                                                                                                      |
| CTRL/L | <p>Requests form feed.</p>                                                                                                                                                                                                                                                                                                                           |
| CTRL/O | <p>Alternately suppresses and continues display of data at the terminal.</p>                                                                                                                                                                                                                                                                         |
| CTRL/Q | <p>Restarts terminal output that was suspended by CTRL/S.</p>                                                                                                                                                                                                                                                                                        |
| CTRL/R | <p>Retypes the current line during input and leaves the cursor positioned at the end of the line.</p>                                                                                                                                                                                                                                                |
| CTRL/S | <p>Suspends terminal output until you press CTRL/Q.</p>                                                                                                                                                                                                                                                                                              |
| CTRL/U | <p>Cancels the current line and discards it.</p>                                                                                                                                                                                                                                                                                                     |
| CTRL/X | <p>Cancels the current line and deletes data in the type-ahead buffer.</p>                                                                                                                                                                                                                                                                           |
| CTRL/Y | <p>Interrupts command or program execution and returns control to the command interpreter.</p>                                                                                                                                                                                                                                                       |
| CTRL/Z | <p>Signals end-of-file for data entered from the terminal.</p>                                                                                                                                                                                                                                                                                       |
| TAB    | <p>Moves the printing element or cursor on the terminal to the next tab stop on the terminal. Most terminals have tab stops at every 8-character position on a line.</p>                                                                                                                                                                             |
| DELETE | <p>Deletes the last character entered at the terminal and backspaces over it. On some terminals, this key is labeled RUBOUT.</p>                                                                                                                                                                                                                     |
| ESCAPE | <p>Has special uses for particular commands or programs, but generally performs the same function as RETURN. On some terminals, this key is labeled ALTMODE, ESC, or ESC(SEL).</p>                                                                                                                                                                   |

## 2.7 SYNTAX RULES

VAX/VMS MCR allows you to enter a command on multiple lines, place comments on a command line, and abbreviate command names.

### 2.7.1 Continuing Commands on More Than One Line

You can enter a command string on more than one line by using the continuation character, a hyphen (-), as the last element on a line. For example:

```
> COPY /LOG -
>_ INFIL.DAT -
>_ OUTFIL1.DAT
```

No restriction is placed on the number of lines you can use to enter a command string.

After each continued line, the command interpreter prompts for additional input (>\_).

### 2.7.2 Using Comments in Command Lines

VAX/VMS MCR distinguishes command lines from comments using conventions of RSX-11M-PLUS, that is:

- A semicolon (;) as the first significant character on a line indicates that the entire line is a comment. A full-line comment cannot be continued with the hyphen (-).
- An exclamation mark (!) usually indicates the beginning of a comment that ends a command line. An exclamation mark (!) can also indicate the beginning or end of a comment embedded within the actual command text. That is, if all or part of a command will follow a comment, place a second exclamation mark to indicate where the comment ends and the command text begins. Embedded comments and the command they describe can be continued over multiple lines. For example:

```
>DIRECTORY/FULL !VIEW FILE PROTECTION FOR THESE-
>_ FILES!AVDEGREE.DAT,!DEGREE DAY DATA AVERAGE!-
>_ CURDEGREE.DAT!THIS YEAR'S ACTUAL DATA
```

### 2.7.3 Truncating Command Names and Keywords

Most command names can be truncated to their first three characters. You can truncate command names to fewer than three characters as long as the name remains unique. For example, because the TYPE command is the only command beginning with the letters TY, it has a minimum truncation of two letters. For convenience, the frequently used RUN command can be truncated to one letter.

However, you must type the first four characters of some commands in order for VAX/VMS MCR commands to be distinguished from RSX-11M command names which contain the same initial three characters. These exceptions are noted in the command descriptions in Chapter 4.

All other portions of a command line (for example, keywords and keyword values) can be truncated to four letters, or fewer, if they remain unique within the command.

#### 2.7.4 Placing Keywords

VAX/VMS MCR is more flexible than RSX-11M MCR about where you can place keywords in the command string. The command descriptions in Chapter 4 separate command keywords from file specification keywords. Command keywords have the same meaning whether they appear after the command name or after a command parameter. For example, the following two commands are identical:

```
> RUN /UIC=[200,230] DOTS.TSK
> RUN DOTS.TSK /UIC=[200,230]
```

File specification keywords, on the other hand, have different meanings, depending on where they are placed in the command string. If specified immediately after a file specification, they affect only the file thus qualified. If specified after the command name, they affect all files specified as parameters. For example, the first command below results in the printing of two copies of both files, whereas the second results in the printing of two copies of SANDPIPER.MAP and one copy of YELLOWLEG.MAP.

```
> PRINT/COPIES=2 SANDPIPER.MAP, YELLOWLEG.MAP
> PRINT SANDPIPER.MAP/COPIES=2, YELLOWLEG.MAP
```

Some file specification keywords are not valid as command keywords. Individual keyword descriptions indicate any restrictions.

#### 2.7.5 Specifying File Protection

All disk and tape volumes have protection codes that restrict access to the volume. This protection is assigned when the volume is initialized and can be reassigned when a volume is mounted.

For disk volumes, each file on the volume, including the directory file, can have a different protection associated with it.

Files potentially can be accessed by users in four categories:

- SYSTEM -- all users who have low group numbers, usually 1 to 10 (octal) or users who have SYSPRV (or BYPASS) privilege. However, the exact range of group numbers is determined by the system manager when the system is generated and may range from 1 to 377. These group numbers and privileges are generally for system managers, system programmers, and operators.
- OWNER -- the UIC of the person who created and, therefore, owns the volume or file.
- GROUP -- all users who have the same group number in their UICs as the owner of the file.
- WORLD -- all users who do not fall into any of the other three categories.

Each of these categories of user can be allowed or denied the following types of access:

- READ -- the right to examine, print, or copy a file or files on a volume.
- WRITE -- the right to modify the file or to write files on a volume.
- EXECUTE -- the right to execute files that contain executable program images. When applying protection to an entire volume, this field is interpreted as the right to create files on the volume.
- DELETE -- the right to delete the file or files on the volume.

Any combination of access types can be specified for any category of user.

When you specify a protection code, you must abbreviate protection types to one character; you can specify the user categories and protection types in any order. If you omit a protection type for a user category, that category of user is denied that type of access. When specifying file protection, if you omit a user category, the current access rights for that category remain unchanged. When you specify volume protection, omission of a category denies that user category all access.

When you specify a protection code, separate each user category from the access type with a colon. If you want to deny all access to a category, specify the category without the trailing colon, for example, G. To specify more than one user category, separate each category with a comma and enclose the entire code specification in parentheses, as follows.

```
SET PROTECTION=(SYSTEM:RWED,GROUP:R,WORLD)/DEFAULT
```

This protection code allows the system all types of access; allows group members read access only; prohibits all access by users in the world category; and does not change the current default for the file's owner.

### 2.7.6 Entering Dates and Times

When a command accepts a keyword that specifies a time value, the time value is either an absolute time or a delta time:

- An absolute time is a specific date and time of day (for example, 20-JUN-1982 10:53:22.10).
- A delta time is a future offset from the current date and time of day (for example, two days and five hours from now).
- An absolute and delta time combination consists of an absolute time value plus or minus a delta time value, for example, tomorrow plus one day.

**2.7.6.1 Absolute Time** - An absolute time generally has the following format:

```
[dd-mmm-yyyy[:]][hh:mm:ss.cc]
```

## VAX/VMS ENVIRONMENT

You can specify the date, the time, or both. The variable fields are as follows:

| Field | Meaning                                                                                                                                                    |
|-------|------------------------------------------------------------------------------------------------------------------------------------------------------------|
| dd    | Day of month (1 to 31)                                                                                                                                     |
| mmm   | Month; the month must be specified as one of the following 3-character abbreviations:<br><br>JAN, FEB, MAR, APR, MAY, JUN,<br>JUL, AUG, SEP, OCT, NOV, DEC |
| yyyy  | Year                                                                                                                                                       |
| hh    | Hour of the day (0 to 23)                                                                                                                                  |
| mm    | Minute of the hour (0 to 59)                                                                                                                               |
| ss    | Seconds (0 to 59)                                                                                                                                          |
| cc    | Hundredths of seconds (0 to 99)                                                                                                                            |

You may also specify one of the following keywords whenever an absolute time is appropriate:

| Keyword   | Description                                            |
|-----------|--------------------------------------------------------|
| TODAY     | The current day, month and year at 00:00:00:0 o'clock. |
| TOMORROW  | 24 hours after 00:00:00:0 o'clock today.               |
| YESTERDAY | 24 hours before 00:00:00:0 o'clock today.              |

The punctuation marks indicate how the system interprets the time value you enter:

- If you specify both the date (dd-mmm-yyyy) and the time (hh:mm:ss.c), you must type a colon between the date and the time.
- You can truncate the date and the time on the right; however, if you are specifying both a date and a time, the date format must contain at least one hyphen.
- You can omit any of the fields within the date or time, as long as you type the punctuation marks; the system supplies default values.
- The period between seconds and hundredths of seconds is a delimiter; it is not a decimal point.

When you omit the date or any of its fields from an absolute time value, the system supplies the current day, month, and year by default.

When you omit any fields from the time, the system supplies a value of 0 for the field.

## Examples:

| Time Specification | Result                                                  |
|--------------------|---------------------------------------------------------|
| 28-JUN-1982:12     | 12:00 noon on June 28, 1982                             |
| 28-JUN             | Midnight (00:00 o'clock) on the 28th of June, this year |
| 15                 | 3:00 P.M., today                                        |
| 15-                | The 15th day of the current month and year, at midnight |
| 18:30              | 6:30 P.M., today                                        |
| 15--:::30          | 00:30 o'clock, on the 15th day of the current month     |

When you specify an absolute time that has already passed, the system performs the request immediately.

### 2.7.6.2 Delta Time - A delta time has the format:

[dddd-][hh:mm:ss.cc]

The variable fields are as follows:

| Field | Meaning                                   |
|-------|-------------------------------------------|
| dddd  | Number of days, 24-hour units (0 to 9999) |
| hh    | Number of hours (0 to 23)                 |
| mm    | Number of minutes (0 to 59)               |
| ss    | Number of seconds                         |
| cc    | Number of hundredths of seconds (0 to 99) |

When you specify a delta time value, you can truncate the time field on the right; you can also omit any of the variable fields, as long as you supply the punctuation marks. When any field is omitted from a delta time value, the system supplies a value of 0 for the field.

## Examples:

| Time Specification | Result                           |
|--------------------|----------------------------------|
| 3-                 | 3 days from now (72 hours)       |
| 3                  | 3 hours from now                 |
| :30                | 30 minutes from now              |
| 3-:30              | 3 days and 30 minutes from now   |
| 15:30              | 15 hours and 30 minutes from now |

2.7.6.3 Absolute and Delta Time Combinations - An absolute and delta time combination is an absolute time plus (+) or minus (-) a delta time value. If you omit the absolute time value, the delta time is offset from the current date and time of day. The delta time value must always be preceded by a plus or minus sign.

The variable fields and default fields for absolute time values are the same as those described in the preceding sections.

Whenever a plus sign (+) precedes the delta time value, the entire time specification must be enclosed in quotes ("). However, if a minus sign (-) is used, no quotes are needed.

Examples:

| Time Specification  | Result                       |
|---------------------|------------------------------|
| " +5"               | Current time plus 5 hours    |
| " +:5"              | Current time plus 5 minutes  |
| " TOMORROW+1-00:00" | Tomorrow plus 1 day          |
| -:5                 | Current time minus 5 minutes |

## 2.8 ERROR MESSAGES

Under VAX/VMS, when you enter an MCR command incorrectly, the command interpreter issues a message in the standard VAX/VMS format. These messages do not look like RSX-11M MCR error messages. Rather, they have the following general format:

%XXX-L-CODE, text

XXX is a mnemonic for the operating system program issuing the message.

L is a severity level indicator: S for success; I for information; W for warning; E for error; and F for fatal.

CODE is a shorthand code for the message text that follows the code.

For example:

%MCR-W-MAXPARAM, maximum parameter count exceeded

Because VAX/VMS messages are descriptive, you can usually learn the corrective action from them. The VAX/VMS System Messages and Recovery Procedures Manual lists all system error messages and describes corrective measures.

RSX-11M components (for example, PIP and TKB) issue the same messages under VAX/VMS that they issue under RSX-11M. When using VAX/VMS MCR to run RSX-11M components, you should consult the appropriate RSX-11M documentation for error message information. For example, if you encounter an error message while using PIP, see the RSX-11M/M-PLUS Utilities Manual for an explanation of the message.

## CHAPTER 3

### USING VAX/VMS MCR

Using the VAX/VMS MCR interface, you can perform three types of tasks:

- Run RSX-11M task images and VAX-11 images
- Use RSX-11M components for RSX-11M program development (for example, MACRO-11 or Task Builder)
- Execute RSX-11M indirect command files (for example, RSX-11/S system generation indirect command files)

This chapter explains how to log in to the VAX/VMS system and perform the types of tasks listed above. In addition, this chapter describes considerations for task building RSX-11M task images that are to run under VAX/VMS.

#### 3.1 LOGGING IN TO VAX/VMS

There is no LOGIN command. Rather, you get the attention of the system from a terminal that is currently not in use by pressing the RETURN key (or press CTRL/C or CTRL/Y).

The system prompts for your user name and then for your password:

Username:

Password:

When you enter your password, the system does not display it.

Using your entry in the user authorization file, the system verifies your right to gain access to the system and sets up the default characteristics for the terminal session. It then executes the login indirect command file specified in your user authorization file entry, unless you specified a different indirect command file or requested that the login file not be executed when you logged in. If a login indirect command file is not specified there, login searches your default directory for an indirect command file named LOGIN.CMD and executes this file if it is present. Finally, the command interpreter prompts for command input.

### 3.1.1 Selecting a Command Interpreter

The system determines which command interpreter is to be associated with your process using the following factors:

- The default command interpreter named in your user authorization file entry
- The /CLI keyword that you specify after your user name to override the default command interpreter for the terminal session

If the MCR command interpreter is not your customary default, then you must enter your user name followed by /CLI=MCR, as follows:

Username: CALEB/CLI=MCR

Then, enter your password in the normal manner.

Note that the VAX/VMS system does not support MCR indirect command files unless your command interpreter is MCR. Characteristics about DCL and MCR that you may consider when choosing a command language are discussed in Sections 1.2 and 1.3.

Also, once you log in with MCR as your command interpreter it remains in control of your process until you log out. The DCL command interpreter cannot be accessed when you are logged in with the VAX/VMS MCR command interpreter.

### 3.1.2 LOGIN.COMD File

The LOGIN.COMD file used under VAX/VMS is identical in function to a LOGIN.COMD file under RSX-11M. If your default command interpreter is MCR, or if you request MCR using the /CLI keyword to login, the MCR command interpreter searches your default directory for the file LOGIN.COMD. The LOGIN.COMD must contain valid VAX/VMS MCR commands and directives.

## 3.2 PREPARING TO RUN AN IMAGE

The basic steps required to prepare an image for execution are the same for VAX-11 and RSX-11M images.

- Allocate and mount any physical devices that the image requires.
- Equate VAX/VMS physical device names to any logical device names or RSX-11M physical device names that the image uses.

In order for an image to be executed, it must reside on a Files-11 disk volume. The volume must be mounted and you must have the right to gain access to it (that is, you must be allowed execute (E) access as described in Section 2.7.5).

### 3.2.1 Allocating Devices

Device allocation under VAX/VMS is essentially the same as device allocation under RSX-11M, that is, you can use the ALLOCATE command to reserve a device, usually before you mount it. Once allocated, the device is not available to other users. The device remains allocated

until you deallocate it or log out of the system. Under RSX-11M, allocated devices are reserved for use by any tasks that you initiate. Under VAX/VMS, on the other hand, allocated devices are reserved for any image that runs in your current process or in one of its subprocesses.

The VAX/VMS ALLOCATE command accepts as a parameter either a logical device name or a VAX/VMS physical device name. If you specify a physical device name, it can be either explicit or generic.

**3.2.1.1 Generic Device Names** - A generic device name explicitly specifies the device type and, optionally specifies the controller or unit. A generic device name allows the system to select the actual device unit to be allocated from available devices of a specific type. For example, the following command requests VAX/VMS to allocate any available TE16 tape drive:

```
> ALL MT:
 _MTB2: ALLOCATED
```

The system response indicates that drive 2 on controller B has been allocated. When the device type and controller are specified in a generic device name, the ALLOCATE command looks for an available unit on a specific controller. Similarly, when the device type and unit number are specified, the ALLOCATE command looks for a controller of the designated type that has the specific device unit available.

Unlike RSX-11M, when the controller and unit number are omitted, VAX/VMS does not default to unit 0 of controller A. This difference may affect indirect command files that allocate devices expecting unit 0 to be used by default; refer to Chapter 5 in this manual.

Refer to the VAX/VMS Magnetic Tape User's Guide for information about using magnetic tape units. The VAX/VMS Command Language User's Guide provides additional information about allocating disk units.

### 3.2.2 Mounting Volumes

Volumes are mounted in essentially the same way under VAX/VMS as they are under RSX-11M; both systems support a MOUNT command. Both systems define similar functions requested by specifying MOUNT command keywords. The actual keyword names, however, differ between systems. The VAX/VMS MOUNT command also has additional options that function as follows:

- Define a logical name for the device on which the volume is mounted
- Specify volume owner and volume protection
- Specify the degree of sharing that is allowed on the volume (for example, specify a private volume or one that is available to the group or world)
- Specify whether operator or user intervention is allowed.

All of the MOUNT command options are described in Chapter 4 in the MOUNT command description.

The main purpose of the MOUNT command is to gain access to the volume. However, under VAX/VMS you can use certain disk volumes without issuing a MOUNT command for them. Any volume that another user (most

likely, the system manager or operator) has mounted specifying the /SYSTEM keyword is available to anyone logged in to the system. Any volume that another user has mounted specifying the /GROUP keyword is available to all users with the same UIC group number as the user who issued the MOUNT command.

The MOUNT command accepts either a logical device name or a physical device name as a parameter. If you specify a physical device name, you can do so either explicitly or by using defaults. The default for the controller designator is A, and the default for unit number is 0.

The VAX/VMS Command Language User's Guide and the VAX/VMS Magnetic Tape User's Guide provide additional information about mounting disk and tape volumes.

### 3.2.3 Assigning Logical Names

Under VAX/VMS, you can use logical names in RSX-11M task images, VAX-11 native mode images, and MCR commands. Using the ASN (Assign) command, you can equate a logical name with a VAX/VMS physical device name, as in the following example:

```
ASN DBB2:=IN0:
```

You also can equate a logical name with all or a portion of a file specification, as in the following example:

```
ASN DBB2:[MYDIR]FILEA.DAT;2=INPUT
```

Unlike RSX-11M logical names, VAX/VMS logical names are not restricted to two letters and a unit number. VAX/VMS logical names can consist of as many as 63 alphanumeric characters. Note that the ASN command can also delete a logical name. The following sections describe the use of logical names for RSX-11M task images and VAX-11 native mode images.

#### 3.2.3.1 Using Logical Names for RSX-11M Task Images - RSX-11M task images issue ASSIGN LUN directives to assign a logical unit number (LUN) to an RSX-11M physical device unit or a logical name. Similarly, the RSX-11M Task Builder ASG option can be used to assign a LUN to a physical or logical device at build time.

When either method is used to assign a LUN to an RSX-11M physical device unit, you can use the ASN (Assign) command to define the RSX-11M physical device name as the logical name for a VAX/VMS device. Then, when VAX/VMS encounters the RSX-11M name in the program, it translates the name to the associated VAX/VMS physical device name. The following example illustrates the use of the ASN command.

```
ASN DBA1:=DK2:
```

When the ASSIGN LUN directive is issued or when the image activated has a task builder option to assign a LUN to a logical-device name, you can use the ASN command to equate the logical name with a VAX/VMS physical device name. For example:

```
ASN DBB0:=XY:
```

The logical name assignment can be made by entering the appropriate ASN command before entering the RUN command to invoke the RSX-11M task

image. Alternatively, an indirect command file can execute both the ASN command and RUN command, thus reducing the possibility of operator error at run time.

If you do not assign logical names, VAX/VMS attempts to map the RSX-11M device names used by the image to VAX/VMS device names, as described in Section 2.5.2.

**3.2.3.2 Using Logical Names Instead of the RSX-11M MCR Command REA -** Under RSX-11M, a privileged user can assign a physical or logical device name to a LUN using the REA (Reassign) command.

Under VAX/VMS, you must assign LUNs at either the program or task-build level; the VAX/VMS MCR command interpreter does not support the REA command. However, the logical name facility provides the flexibility needed to associate the RSX-11M device name used in an ASSIGN LUN directive or ASG option with a VAX/VMS physical device, as described above.

**3.2.3.3 Using Logical Names Instead of Redirect -** Under RSX-11M, a privileged user can redirect I/O operations from one device to another during task execution. VAX/VMS does not support the RED (Redirect) command; you must assign logical names before running the image.

**3.2.3.4 Using Logical Names for VAX-11 Native Images -** The VAX/VMS logical name capability provides more flexibility than the RSX-11M capability. Native images can use a logical name that is equated with all or part of a file specification starting with the leftmost item of the file specification. Similarly, you can use logical names that are equated to file specifications at the command level. The following is an example of using logical names:

```
> ASN DBB1:[JONES]SNOW.DAT;4=INPUT
> RUN ANLZE
.
.
.
```

The ASN (Assign) command equates the logical name INPUT with the full file specification DBB1:[JONES]SNOW.DAT;4. When the image ANLZE refers to INPUT, VAX/VMS automatically translates it to the associated file specification.

The full VAX/VMS logical name capability is available through VAX/VMS MCR; the VAX/VMS Command Language User's Guide provides detailed information on logical names.

### 3.2.4 Installing RSX-11M Task Images

Under VAX/VMS, images do not have to be installed before execution. You run images under VAX/VMS in the same manner that a nonprivileged user runs them under RSX-11M, that is, by supplying the file specification of the image as a parameter to the RUN command. VAX/VMS does not support the RSX-11M INSTALL command.

You can use logical names to create an effect similar to that of the INSTALL command. If you assign a logical name to the file

## USING VAX/VMS MCR

specification of the image file, the logical name serves as a task name. If you type the logical name in response to the MCR prompt (>), VAX/VMS performs logical name translation and runs the associated image.

For example, the following command establishes REP as the logical name for the image file DB4:[IMAGES]ALPHA.TSK contained on DBA4.

```
ASN DB4:[IMAGES]ALPHA.TSK=REP/GBL
```

Because the logical name REP is placed in the system logical name table, any user can type REP to run the image, as follows:

```
REP additional-data
```

The image can obtain the additional data by issuing a GET MCR COMMAND LINE directive, just as it does in RSX-11M.

Note that for a process to place names in the system logical name table, the user must have the SYSNAM privilege.

### 3.3 RUNNING IMAGES

The VAX/VMS MCR command interpreter lets you run VAX/VMS and RSX-11M images. The basic format of the command to request image execution is as follows:

```
RUN file-spec
```

The parameter file-spec represents a standard VAX/VMS file specification or a logical name that translates to one. If the file type is not supplied, it defaults to EXE. To run RSX-11M task images having a file type of TSK, you must either explicitly supply the file type, or you must rename the image files. EXE is the default file type for image files produced by the VAX-11 Linker and the VAX/VMS version of the RSX-11M Task Builder.

The RUN command provides keywords that control whether the requested image is run in your process, a subprocess, or a detached process. Additional keywords allow you to control the process's priority, scheduling, resource quotas, and privileges. A detailed description of the RUN command and all its options can be found in Chapter 4.

The following are examples of the RUN command:

```
> RUN INVENTORY
```

This command runs the image in the highest version of the file INVENTORY.EXE located on the default disk device and directory.

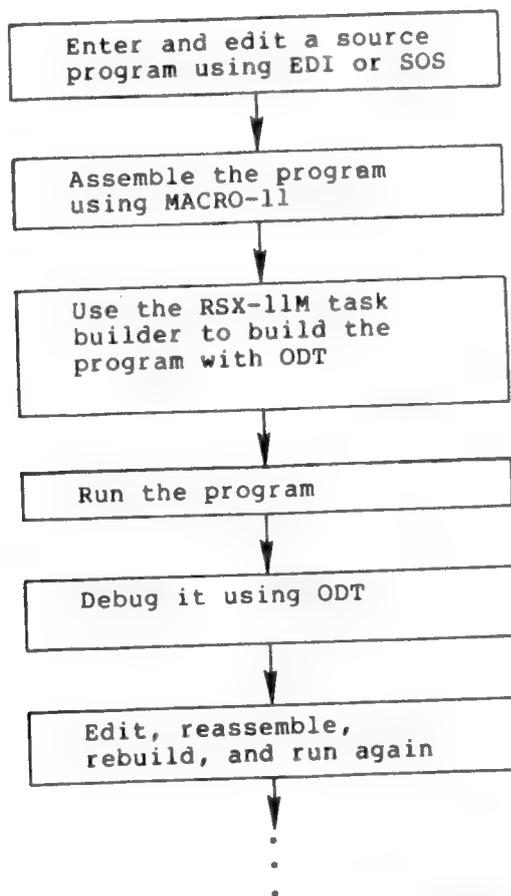
```
> ASN DBB2:[RSX]IMAGE.TSK;8=MYTASK
> RUN MYTASK
```

This sequence assigns MYTASK as the logical name for the image. The RUN command translates MYTASK to DBB2:[RSX]IMAGE.TSK;8 and runs it.

Instead of using the RUN command, you can run images in your process by entering the file specification of the image file in response to the MCR prompt (>), or by entering a logical name that translates to a file specification.

3.4 PROGRAM DEVELOPMENT UNDER VAX/VMS MCR

VAX/VMS MCR permits the development of RSX-11M task images. See Section 1.2 for a complete list of program development utilities available under VAX/VMS MCR. Figure 3-1 illustrates the general flow of RSX-11M (compatibility mode) program development that you can perform using VAX/VMS MCR.



ZK-900-82

Figure 3-1: RSX-11M Program Development Cycles Using VAX/VMS MCR

3.4.1 Using RSX-11M Components

You can use any of the RSX-11M components listed in Section 1.2.1 by issuing a request to the VAX/VMS MCR command interpreter. To request any of these components, either enter the component name and press RETURN, or enter the component name followed by a valid command line for that component and press RETURN. Subsequent operation of the component is identical to that under RSX-11M, as documented in such RSX-11M manuals as the IAS/RSX-11 ODT Reference Manual, the RSX-11M/M-PLUS Task Builder Manual, and the RSX-11M/M-PLUS Utilities Manual.

In addition, you can build ODT into any RSX-11M task image that requires debugging. ODT operates just as it does under RSX-11M. You also can interrupt image execution and type the DEBUG command to pass control to ODT if the image was built with ODT.

## USING VAX/VMS MCR

Of the RSX-11M components available, all accept the standard VAX/VMS file specifications except FLX. Considerations for using FLX are provided in Section 3.4.2.

VAX/VMS does not support the PIP/UF switch; you must instead use the MCR command UFD (see Chapter 4) or the DCL command CREATE/DIRECTORY (see the VAX/VMS Command Language User's Guide).

The following are examples of invoking RSX-11M components from VAX/VMS MCR:

```
> EDI
EDI> DBA1:[COLD]CENTIGRAD.DAT
[CREATING NEW FILE]
INPUT
.
.
.
<RET>
*EX
[EXIT]
>
```

Typing EDI in response to the MCR prompt (>) and pressing RETURN causes EDI to prompt for a command line. EDI operates just as it does under RSX-11M, that is, the default directory is searched for the file CENTIGRAD.DAT. Failure to locate the file results in creation of a new file under the directory [COLD]. Exiting from EDI causes MCR to prompt.

```
> MAC @MACINP
>
```

This command requests the MACRO-11 Assembler to read the indirect command file MACINP.CMD for command input. After the indirect file is processed, MCR prompts again.

```
> TKB APPLES=GRAPES
>
```

Typing TKB followed by a command line results in the task building of the object file GRAPES.OBJ on the default disk and directory to produce the image file APPLES.EXE, which is also stored on the default disk and directory. GRAPES.OBJ must have been produced by an RSX-11M compiler or assembler. When task building completes, VAX/VMS MCR prompts again.

```
> TKB
TKB> AVERAGE,AVERAGE=AVERAGE
TKB>/
ENTER OPTIONS:
TKB>
.
.
.
TKB>//
>
```

The sequence above invokes the RSX-11M Task Builder, supplies it with the names for the input and output files, and requests the task builder to prompt for options. The two slashes cause the Task Builder to exit; MCR prompts.

## USING VAX/VMS MCR

```
> PIP DBB2:[ADAMS]*.LST;*/DE
>
```

The command above invokes PIP to delete all listing (LST) files from directory [ADAMS] on DBB2. VAX/VMS MCR prompts after the files are deleted.

```
> PIP
PIP> SPICE.*/PU
PIP> .
.
.
PIP> ^Y
>
```

In the above sequence, typing PIP in response to the MCR prompt causes PIP to prompt. PIP reprompts after each command until you press CTRL/Y to pass control to the MCR command interpreter. Note that PIP displays listed file version numbers in decimal rather than octal.

### 3.4.2 Using the File Transfer Program

The File Transfer Program (FLX) interprets only RSX-11M file specifications, that is, file specifications with the following format:

```
ddu:[g,m]filename.type;version
```

FLX cannot interpret a controller designation in the device name or a directory name (other than the [g,m] form) in the file specification; nor does it accept a logical name for a file specification.

If you omit the directory from a file specification for a disk file, VAX/VMS uses your default directory. It can be in either the [g,m] form or a directory or subdirectory name.

Because FLX does not accept VAX/VMS device names, your use of FLX depends on the device mapping that the operating system performs automatically. Device mapping is described in Section 2.5.

If you specify a device name that does not map to a physical device on VAX/VMS, FLX issues an error message indicating that it could not find the device.

Before using FLX with a magnetic tape, you must allocate the tape drive. Unless you are using an ANSI-labeled magnetic tape, you must mount the tape drive after you allocate it using the /FOREIGN keyword for the MOUNT command. DOS-11 and RT-11 tapes are not ANSI-labeled and thus require the MOUNT/FOREIGN command. Also, note that ANSI (17-character) file names on magnetic tape are supported under VAX/VMS compatibility mode as described in the VAX/VMS Magnetic Tape User's Guide.

The following are examples of FLX command lines.

```
> FLX DB0:[120,30]=MT1:SYS1.MAC/DO
```

VAX/VMS maps the device name DB0 to the VAX-11 physical device DBA0. It maps device name MT1 to device MTA1. File SYS1.MAC is transferred to directory [120,30] on DBA0.

```
> FLX
FLX> DR3:=DM0:[133,10]MYFIL.MAC/RT
```

## USING VAX/VMS MCR

VAX/VMS maps device name DR3 to the VAX-11 physical device DRA3 and DMO to DMA0. The user's default directory on DRA3 contains the output file.

See the VAX-11 Utilities Reference Manual for further information on FLX.

### 3.4.3 Using VAX/VMS Components

RSX-11M users who are not familiar with VAX/VMS utilities should be aware of certain differences between RSX-11M and VAX-11 utilities. Under RSX-11M, some optional parameters are specified by using "switches." Certain switches allow a minus sign (-) to precede the switch name to indicate that switch is to be turned off, or negated, usually to override a default. Under RSX-11M MCR, the switches and any required parameters are usually specified on the command line following the command name.

Under VAX/VMS, these optional parameters are called "qualifiers," not switches. To negate a qualifier (switch) while using a VAX/VMS utility or command, precede the qualifier with the word NO instead of a minus sign (-). With required parameters, most VAX/VMS utilities and certain MCR commands that are identical to DCL commands will prompt for the required parameter if it was omitted from the entered command line. You can omit one or more required parameters to cause prompts to appear, for instance, if you are not sure of the order in which the parameters must be entered. Like an RSX-11M switch, each utility qualifier must be preceded by a slash (/).

The following information describes how to invoke the VAX/VMS utilities available under MCR:

- Disk Save and Compress Utility for Files-11 Structure Level 2 volumes -- type DSC2 followed by any qualifiers needed. DSC2 prompts you for a command.
- EDT Editor -- type EDT followed by any qualifiers needed and optionally a file specification. EDT prompts you for the file to be edited if one was not specified. (To invoke EDT under DCL, type EDIT instead of EDT.)
- Personal Mail Utility -- type MAIL followed by any qualifiers needed. MAIL prompts you for a command.
- SOS Editor -- type SOS followed by the required file specification and any qualifiers needed. SOS prompts you for the file specification if one was not specified; include any qualifiers needed with the file specification entry.
- VAX-11 Symbolic Debugger -- after interrupting a native mode image, type DEBUG (see Chapter 4 for additional information).

For information about the qualifiers and operation of these utilities, see the appropriate VAX/VMS document. To determine which document best describes a particular utility, see the VAX-11 Information Directory and Index.

### 3.5 EXECUTING RSX-11M INDIRECT COMMAND FILES

RSX-11M indirect command files have the same uses under VAX/VMS as under RSX-11M. That is, you can specify an indirect command file for processing by VAX/VMS MCR or you can specify an indirect command file

in a command to an RSX-11M component executing under VAX/VMS. In either case, an "at sign" (@) precedes the file specification that identifies the indirect file just as it does under RSX-11M.

Indirect command files specified for use by RSX-11M components (for example, Task Builder and MACRO-11) are identical to those used under RSX-11M. Section 3.1.1 describes the login procedure used to establish MCR as your command interpreter; which is necessary for support of indirect command files. Chapter 5 describes the MCR indirect command file processor directives available under VAX/VMS and provides additional information about running MCR indirect command files. The Execute Procedure (@) is described in Chapter 4.

### 3.6 BUILDING RSX-11M IMAGES FOR EXECUTION UNDER VAX/VMS

The VAX/VMS system can serve as the host for (1) modifying existing RSX-11M task images so that they execute under VAX/VMS in compatibility mode or (2) developing new task images in compatibility mode to run under RSX-11M. When the task image being modified or developed is to execute under VAX/VMS, you should build it with that fact in mind.

Programs developed under VAX/VMS MCR fall into one of two groups:

- Programs that will execute under VAX/VMS in compatibility mode (or RSX-11M). See Section 1.1.1 and the following sections for restrictions that apply to programs in this group. To develop programs in this group, use the RSX-11M program development utilities, such as MAC (MACRO-11) and TKB (Task Builder).
- Programs that will execute only under RSX-11M, not under VAX/VMS. No restrictions apply to this group of programs, except for modules that do not observe the restrictions listed in Section 1.1.1 and the following sections allow limited debugging under VAX/VMS, and should be debugged under RSX-11M. To develop programs in this group, use the RSX-11M program development utilities, such as MAC (MACRO-11) and TKB (Task Builder).

#### 3.6.1 Task Builder Switches

You can use most of the switches defined for the RSX-11M Task Builder when building an image for execution under VAX/VMS. One RSX-11M Version 3.2 switch you cannot use is -MM (unmapped system); that is, the task image must be built to execute on a mapped system. New Task Builder switches added since RSX-11M Version 3.2 are not likely to be supported under VAX/VMS compatibility mode, although they may be simply ignored.

Table 3-1 lists the switches that are known to be effective under VAX/VMS for RSX-11M Version 3.2. With the exception of -MM, unlisted switches have no effect under VAX/VMS and do not cause an error when the task image executes under VAX/VMS. As a general rule, switches that are intended to affect the way RSX-11M treats the task image during execution are ignored. Switches requiring privilege are also ignored.

Defaults and the use of a minus sign (-) to negate switches are identical to RSX-11M.

## USING VAX/VMS MCR

Use of overlays can be avoided to take advantage of VAX/VMS virtual address space. VAX/VMS provides 65K-bytes of virtual address space for RSX-11M images.

### 3.6.2 Task Builder Options

You can specify any of the RSX-11M Task Builder options for RSX-11M task images that are to run under VAX/VMS. The options that specify user identification code (UIC), priority, and partition have no effect; UIC and priority are associated with the user running the task image under VAX/VMS, and partitions do not exist. Two options, TASK and ASG, may be required for RSX-11M task images to run successfully under VAX/VMS, as described below.

RSX-11M task images that use common event flags (that is, specifying event flag numbers greater than 33), that are the target of directive requests, or that issue RECEIVE DATA and RECEIVE DATA OR EXIT directives must have a name assigned using the TASK option. VAX/VMS uses the presence of a task name in the image label block as an indication that the image requires special initialization.

RSX-11M task images that do not associate LUNs with either a physical device name or a logical device name within the program must be built using the ASG option to make the required association. VAX/VMS MCR does not support the REA (Reassign) command.

### 3.6.3 Noncontiguous Task Image Files

The Task Builder issues an informational message if it produces a task image file that is not contiguous. VAX/VMS, however, allows you to run task images contained in noncontiguous files.

### 3.6.4 Building RSX-11M Task Images on VAX/VMS for Execution on RSX-11M

You also can use the VAX/VMS system to build RSX-11M task images that are to run under RSX-11M. In this case, VAX/VMS imposes no restrictions on the switches and options that you can specify.

Table 3-1: Valid RSX-11M Version 3.2 Task Builder Switches

| Switch or Option | Function                                                                |
|------------------|-------------------------------------------------------------------------|
| CC               | Input file consists of concatenated object modules                      |
| DA               | Image contains a debugging aid                                          |
| DL               | Specified library is a replacement for the system object module library |
| FP               | Task uses the floating-point processor (FPP only)                       |

(continued on next page)

Table 3-1 (Cont.): Valid RSX-11M Version 3.2 Task Builder Switches

| Switch or Option | Function                                                                                                                                               |
|------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| FU               | All co-tree overlay segments are searched for matching definition or reference when modules from the default object module library are being processed |
| HD               | Image includes a header                                                                                                                                |
| LB               | Input file is a library file                                                                                                                           |
| MA               | Memory allocation output includes information from the file                                                                                            |
| MM               | System has memory management; for an image to run under VAX/VMS, MM must be present either explicitly or by default                                    |
| MP               | Input file contains an overlay description                                                                                                             |
| MU               | Image is separated into shareable and nonshareable portions                                                                                            |
| PI               | Image is position-independent                                                                                                                          |
| PM               | Postmortem dump is requested                                                                                                                           |
| SH               | Short memory allocation file is requested                                                                                                              |
| SP               | Spool map output                                                                                                                                       |
| SQ               | Image program sections are allocated sequentially                                                                                                      |
| SS               | Selective search for global symbols                                                                                                                    |
| TR               | Image is to be traced                                                                                                                                  |
| WI               | Memory allocation file is printed at a width of 132 characters                                                                                         |
| XT:n             | Task Builder exits after n diagnostics                                                                                                                 |



## CHAPTER 4

### MCR COMMANDS

This chapter describes VAX/VMS MCR commands and their keywords in detail. Individual command descriptions note restrictions between VAX/VMS MCR commands and RSX-11M MCR commands. For general information on the VAX/VMS MCR environment, refer to Chapter 2 which provides a summary of the VAX/VMS environment.

Certain MCR commands differ in syntax from their VAX/VMS DCL counterparts for consistency with RSX-11M MCR commands. A few examples follow.

- The only syntax difference between the MCR command ASN and the DCL command ASSIGN is the option of placing an equal sign (=) instead of a space between the device (equivalence) name and the logical name for the ASN command, consistent with general RSX-11M syntax rules.
- For the SET UIC commands, the syntax differs. The MCR command SET /UIC has the following format:

```
SET /UIC=[g,m]
```

The DCL command format follows:

```
SET UIC [g,m]
```

The MCR command allows the slash (/) and equal sign (=), unlike its DCL counterpart that requires only a space between the three components of this command.

#### NOTE

VAX/VMS and RSX-11M utilities are not described in this chapter. Refer to the VAX-11 Utilities Reference Manual for such VAX-11 utilities as MAIL. Refer to the VAX/VMS Primer and the EDT Editor Reference Manual for information about the EDT editor. Refer to the VAX-11 SOS Text Editing Reference Manual for information about the SOS editor.

# ALLOCATE

## 4.1 ALLOCATE

The ALLOCATE command reserves an unmounted shareable device or an unallocated nonshareable device as a private device for the process. After the device is allocated, MCR displays the physical name of the device on the current SYS\$OUTPUT device. Other users cannot gain access to an allocated device

### Format:

```
ALLOCATE device name[:]
```

device-name[:] Specifies the physical or logical name of the device to be allocated. If you specify a physical device name, it can be either a complete name or a generic device name, for example TT. See Section 3.2.1.1 for a discussion of generic device names.

### Examples:

```
> ALLOCATE TTH7
_ TTH7: ALLOCATED
```

This command requests allocation of device TTH7. The ALLOCATE command responds by indicating that the requested device has been reserved for the requesting process. The underscore preceding the displayed device name indicates that no further name translation is required to determine the actual physical device allocated.

```
> ASN MTA2:=MT0:
> ALL MT0
_ MTA2: ALLOCATED
```

The ASN command defines MT0 as the logical name for the physical device unit MTA2. The ALLOCATE command then reserves that device using its logical name.

```
>ALL DM;
_ DMA2: ALLOCATED
```

This command requests allocation of any RK06/RK07 disk drive, whose device names always begin with DM. This demonstrates the use of a generic device name.

### Notes:

1. The ASN command can be used to define a logical device name.
2. Assigning an I/O channel (LUN) to a nonshareable device causes the device to be implicitly allocated. Opening a file causes a channel to be assigned.
3. File-oriented devices (magnetic tape and disks) cannot be allocated if they are mounted.

## MCR COMMANDS

4. If you do not specify the device controller designation and unit number under VAX/VMS, the ALLOCATE command selects an available device. The selected device is not necessarily controller A and unit 0.
5. The form of the ALLOCATE command that is compatible between VAX/VMS and RSX-11M MCR is:

ALL device-name[:]

**APPEND****4.2 APPEND**

The APPEND command adds the contents of one or more specific input files to the end of a specific output file.

**Format:**

```
APPEND [keywords] input file spec [file-keywords][,...]
 output-file-spec [file-keywords]
```

**keywords**                    /ALLOCATION=n (file-keyword)  
                              /[NO]CONTIGUOUS (file-keyword)  
                              /EXTENSION=n (file-keyword)  
                              /FILE MAXIMUM=n (file-keyword)  
                              /[NO]LOG  
                              /[NO]NEW (file-keyword)  
                              /PROTECTION=code (file-keyword)  
                              /[NO]READ\_CHECK (file-keyword)  
                              /[NO]WRITE\_CHECK (file-keyword)

**input-file-spec**           Specifies the name of one or more files to be appended. If you specify more than one input file, separate each file specification with either a comma (,) or a plus sign (+), which are equivalent

                              You can use a wild card character in place of the file name, type, or version field. Then, all files that satisfy the remaining components are appended.

**output file-spec**           Is the name of the output file. You must specify at least one field of the output file specification. For fields that you do not specify, the APPEND command uses the related field of the input file specification.

                              If you specify a wild card character in any field of the output file specification, the APPEND command uses the related field of the input file specification(s).

                              If you specify only the device and directory portions of the output file specification, the APPEND command uses related fields of the input file specification.

**Command Keyword:**

/LOG                           Controls whether the file specification of each file that is appended is displayed.  
 /NOLOG  
   (default)

                              If you specify the /LOG keyword with the APPEND command, the file specifications of the input and output files and either the number of blocks or records that are appended will be displayed. The number of new files created will be displayed at the end of command processing.

## MCR COMMANDS

### File Keywords:

- /ALLOCATION=n** Forces the initial allocation of the new output file to the number of 512-byte blocks specified by n. The value n is interpreted as a decimal number by default.
- /ALLOCATION** is valid only if **/NEW** is specified; that is, the allocation values are applied only if a new output file is actually created.
- /CONTIGUOUS**  
**/NOCONTIGUOUS**  
(default) Indicates whether the output file is to be contiguous; that is, whether it must occupy consecutive physical disk blocks.
- By default, the APPEND command creates an output file in the same format as the related input file. If multiple input files of different formats are appended to a single output file, the output file may or may not be contiguous. To ensure that the output file is contiguous, use the **/ALLOCATION** and **/CONTIGUOUS** keywords and create a new output file.
- /EXTENSION=n** Specifies the number of blocks to be added to the new output file each time it is extended.
- /EXTENSION** implies **/NEW**; that is, the allocation values are applied only if a new output file is actually created.
- /FILE\_MAXIMUM=n** Specifies the maximum number of logical records that the output file can contain. This keyword is valid only for new relative files. If you specify the **/FILE\_MAXIMUM** keyword, the **/NEW** keyword is assumed.
- /NEW**  
**/NONEW** (default) Requests that if a specific output file does not already exist, the APPEND command create one. Because **/NONEW** is the default, the output file must already exist.
- /PROTECTION=code** Defines the protection to be applied to the output file. The protection is specified following the standard rules, described in Section 2.7.5. Any protection attributes not specified are taken from the current protection of the output file or, if a new file is created, from the current default protection.
- /READ\_CHECK**  
**/NOREAD\_CHECK**  
(default) Indicates whether the APPEND command is to read a specific input file(s) twice to verify that all records have been correctly read.
- /WRITE\_CHECK**  
**/NOWRITE\_CHECK**  
(default) Indicates whether the APPEND command is to read the output file after it has been written. This verifies that the file has been successfully appended and that the output file can be read without error.

## MCR COMMANDS

### Examples:

```
> APPEND [DAVIS]JOSEPH.OBJ [HICKS]EDWARD.OBJ
```

This command appends the file JOSEPH.OBJ from directory [DAVIS] to the file EDWARD.OBJ in directory [HICKS] on the default device.

```
> APPEND /LOG/NEW FAST.DAT+SLOW.DAT+NORM.DAT TOTAL.DAT
%APPEND-I-CREATED, DBB1:[MAINT]TOTAL.DAT;1 created
%APPEND-S-COPIED, DBB1:[MAINT]FAST.DAT;2 copied to
DBB1:[MAINT]TOTAL.DAT;1 (10 blocks)
%APPEND-S-APPENDED, DBB1:[MAINT]SLOW.DAT;5 appended to
DBB1:[MAINT]TOTAL.DAT;1 (8 records)
%APPEND-S-APPENDED, DBB1:[MAINT]NORM.DAT;12 appended to
DBB1:[MAINT]TOTAL.DAT;1 (22 records)
```

This command concatenates the files FAST.DAT, SLOW.DAT, and NORM.DAT into a new file named TOTAL.DAT. The input and output files are on the default device and directory.

### Notes:

1. RSX-11M MCR does not support the APPEND command.
2. The APPEND command is a variation of the COPY command.
3. When you use wild card characters to concatenate Files-11 Structure Level 1 files, the input files are copied to the output file in random order. When you use wild card characters to concatenate Files-11 Structure Level 2 files, the APPEND command copies files in alphanumeric order. When wild card characters are used in the version field, files are copied in descending order by version number; that is, files with the same name are copied starting with the highest version number.
4. Additional examples of appending files and information about wild card characters are provided in the VAX/VMS Command Language User's Guide.

## ASN

## 4.3 ASN

The ASN command defines or deletes a logical name assignment for a device in one of three logical name tables: process, group, or system.

## Format for Assigning a Logical Name:

```
ASN [keywords] file-spec[=]logical-name
```

## Format for Deleting a Logical Name:

```
ASN = logical-name [keywords]
```

## Format for Deleting all Logical Names:

```
ASN =
```

```
keywords /GBL
 /GROUP
 /TERM
```

file-spec Indicates part or all of a file specification. When a physical device name is specified, terminate the device name with a colon (:). You can also specify a logical device name or pseudodevice name that translates to a physical device name or file specification.

logical-name Specifies a logical name from 1 through 63 characters.

## Command Keywords:

```
/GBL Indicates that the requested action is to occur in the
 system logical name table.

/GROUP Indicates that the requested action is to occur in the
 group logical name table.

/TERM Indicates that the requested action is to occur in the
 (default) process logical name table. /TERM is the default.
```

## Examples:

```
> ASN MTB2:=MT0:
```

This command assigns the string MT0 as the logical name for device MTB2 in the process logical name table.

```
> SHOW LOGICAL MT0
MT0 = MTB2: (PROCESS)
```

This command requests that the equivalence name associated with logical name MT0 be displayed.

## MCR COMMANDS

### Notes:

1. RSX-11M MCR supports the ASN command.
2. The equal sign (=) between the two name parameters is optional; if you do not specify the equal sign, you must leave a space between the two parameters.
3. RSX-11M does not support the /GROUP keyword.
4. VAX/VMS does not support the /TERM keyword.
5. To assign and delete logical names in the group or system logical name tables, GRPNAM or SYSNAM privilege is required.
6. The SHOW LOGICAL command, described in the VAX/VMS Command Language User's Guide, can be used to display logical names in the process, group, and system tables.
7. Assigning a disk device as SY: is not sufficient to cause the disk to become your default disk. To change your default disk use a SET DEFAULT command.

**BYE****4.4 BYE**

The BYE command terminates an interactive terminal session. The system displays a termination message and performs any necessary clean-up operations, such as terminating the current image if one exists, dismounting any private volumes that remain mounted, and deallocating devices. Finally, it deletes your process and subprocesses, if there are any.

**Format:**

BYE

**Example:**

```
> BYE
MURPHY logged out at 24-JUN-1982 13:33:26.57
```

**Notes:**

1. In VAX/VMS MCR, the LOGOUT command and the BYE command perform the same function.
2. RSX-11M MCR supports the BYE command.

## CANCEL

### 4.5 CANCEL

The CANCEL command requests the system to cancel a wake-up request for a specific process.

A wake-up request can be the result of any of the following:

- The RUN\$ directive (see Notes below)
- The Schedule Wake-up system service
- Any MCR or DCL RUN commands that request time-synchronization, such as the /SCHEDULE keyword for the MCR command RUN.

Cancel does not affect the execution of an active process; it only removes time-based requests from the timer queue.

#### Format:

```
CANCEL [keyword] [process-name]
```

keyword                    /IDENTIFICATION=process-id

process-name               Specifies the name of the process for which wake-up requests are to be canceled. Because process names are implicitly qualified by UIC group number, you can cancel requests only for process within your group by specifying a process name.

#### Command Keyword:

```
/IDENTIFICATION=process-id
```

Specifies the process identification of the process for which wake-up requests are to be canceled. If you specify both a process name and identification, the CANCEL command ignores the process name. If you specify neither, wake-up requests for your current process are canceled. The process-id value is interpreted as a hexadecimal number.

#### Examples:

```
> CAN PROCA
```

This command cancels wake-up requests for a process executing in the same group as the requesting process. If the process is not a subprocess of the requesting process, then GROUP privilege is required.

```
> CAN /IDENTIFICATION=C0025
```

This command cancels wake-up requests for the process whose process identification number is C0025. It can be executing outside the group requesting process; appropriate privilege is required, however.

## MCR COMMANDS

### Notes:

1. You must have GROUP privilege to cancel wake-up requests for another process in the same group and WORLD privilege to cancel wake-up requests for another process in the same system. No privilege is required for a process to issue a cancel request for one of its subprocesses.
2. The STOP command can be used to delete a hibernating process for which all wake-up requests are canceled.
3. You can determine whether a subprocess has completed execution of an image by issuing the SHOW PROCESS command with the /SUBPROCESSES keyword described in the VAX/VMS Command Language User's Guide.
4. VAX/VMS uses the Schedule Wake-up system service to emulate the RSX-11M RUN\$ directive; for additional information, refer to the VAX-11/RSX-11M Programmer's Reference Manual.
5. RSX-11M MCR supports the CANCEL command when the VAX/VMS MCR process name is equal to an RSX-11M task name.

## CONTINUE

### 4.6 CONTINUE

The CONTINUE command resumes execution of a command, program, or indirect command file that was interrupted by any of the following:

- Pressing CTRL/Y or CTRL/C
- A PAUSE request issued by a FORTRAN program
- A .PAUSE directive issued in an indirect command file

When a program is interrupted or pauses, only those actions listed below can be taken if the program is to continue to execute. Use of any actions not listed below will cause termination of the interrupted image.

- EXAMINE and DEPOSIT commands
- ALLOCATE and DEALLOCATE commands
- .SETx indirect command file directive (define a symbol)
- ASN command (assign and delete a logical name)
- SET DEFAULT, SET PROTECTION, SHOW DEFAULT, and SHOW PROTECTION commands (set and show the default directory and the default file protection)
- SHOW TRANSLATION (show the translation of a logical name)
- TIME command (display the time)

When an indirect command file pauses, VAX/VMS places no restrictions on the commands that you can use. Restrictions on the use of the .PAUSE directive are presented in Chapter 5.

#### Format:

```
CONTINUE
```

#### Example:

```
> RUN TEST
.
.
.
^Y
> TIME
24-JUN-1982 14:00:55
> CONTINUE
```

The RUN command initiates execution of the image TEST. Pressing CTRL/Y interrupts TEST. The TIME command causes VAX/VMS to display the date and time. The CONTINUE command then requests resumption of TEST at the point where it was interrupted.

## MCR COMMANDS

### Notes:

1. RSX-11M MCR does not support the CONTINUE command.
2. For convenience, the CONTINUE command can be abbreviated to a single character, C.

## COPY

### 4.7 COPY

The COPY command creates a new file from one or more existing files. This command can be used to perform the following functions:

- Copy one file to another file.
- Concatenate more than one file into a single output file.
- Copy a group of files to another group of files.

#### Format:

```
COPY [keywords] input-file-spec [,...]
 output-file-spec [file-keywords]
```

#### keywords

```
/ALLOCATION=n (file-keyword)
/[NO]CONCATENATE
/[NO]CONTIGUOUS (file-keyword)
/EXTENSION=n (file-keyword)
/FILE MAXIMUM=n (file-keyword)
/[NO]LOG
/[NO]OVERLAY (file-keyword)
/PROTECTION=code (file-keyword)
/[NO]READ CHECK (file-keyword)
/[NO]REPLACE (file-keyword)
/[NO]TRUNCATE (file-keyword)
/VOLUME=n (file-keyword)
/[NO]WRITE_CHECK (file-keyword)
```

#### input-file-spec

Specifies the name of one or more files to be copied. If you specify more than one input file, separate the file specifications with either a comma (,) or a plus sign (+).

You can use wild card characters in place of the file name, type, or version field. Then, all files that satisfy the remaining components are copied. For more information on wild card characters, refer to the VAX/VMS Command Language User's Guide.

#### output-file-spec

Is the name of the output file (the file copied to). You must specify at least one field of the output file specification. For fields that you do not specify, the COPY command uses the related field of the input file specification(s).

If you specify a wild card character in any field of the output file specification, the COPY command uses the related field of the input file specification(s) and does not concatenate the output. For more information on wild card characters, refer to the VAX/VMS Command Language User's Guide.

## MCR COMMANDS

If you specify only the device and directory portion of the output file specification, the COPY command uses related fields of the input file specification(s) and does not concatenate the output.

### Command Keywords:

`/CONCATENATE`  
    (default)  
`/NOCONCATENATE`

When multiple input file specifications are provided, this keyword indicates whether a single output file is to be created from all files that satisfy the input file specification.

By default, the COPY command produces a single output file from multiple input files unless:

- One or more wild card characters are present in the output file specification
- Only the device and directory portions of the output file specification are provided

`/LOG`  
`/NOLOG` (default)

Indicates that the COPY command is to display the file specifications of each input file copied in the following format:

input-file copied to output-file (nn blocks)

When the COPY command is creating a single output file from multiple input files, the first message is followed by messages in the following format:

input-file appended to output-file (nn records)

The COPY command displays either the number of records or the number of blocks copied depending on whether the file is copied on a record-by-record or block-by-block basis.

At the end of command processing, the COPY command displays the number of files created.

### File Keywords:

`/ALLOCATION=n`

Forces the initial allocation of the output file to the number of blocks specified by n. If this keyword is not specified, the initial allocation of the output file is determined from the size of the input file being copied.

`/CONTIGUOUS`  
`/NOCONTIGUOUS`

Indicates whether the output file is to be contiguous; that is, whether it must occupy consecutive physical disk blocks.

By default, the COPY command creates an output file in the same format as the corresponding input file. If the input file is contiguous, the COPY command attempts to create a contiguous output file, but does not report an error if the output file is not contiguous.

## MCR COMMANDS

If multiple input files of different formats are copied into a single output file, the output file may or may not be contiguous. To ensure that the output file is contiguous, use the /ALLOCATION and /CONTIGUOUS keywords.

The /CONTIGUOUS keyword has no effect when you copy files to or from tapes because the size of the input file cannot be determined until the file is read. If you copy a file from tape, use the COPY command twice: once to copy the file from tape, and again to create a contiguous file.

- /EXTENSION=n Specifies the number of blocks to be added to the output file each time it is extended. If this keyword is not specified, the default extension attribute of the output file is determined from the input file.
- /FILE\_MAXIMUM=n Specifies the maximum number of logical records that the output file can contain. This keyword is valid only for relative files.
- /OVERLAY  
/NOOVERLAY  
(default) Indicates whether data in the input file is to be copied into an existing output file overlaying the existing data. If the new file is shorter than the overlaid file, the file is truncated at the end of the new file. The new file is in the same position as the overlaid file.
- /OVERLAY is ignored if the output file is written to a non-file-structured device.
- /PROTECTION=code Defines the protection to be applied to the output file. The protection is specified following the standard rules, described in Section 2.7.5. Any protection attributes not specified are taken from the current protection of the input file.
- /READ CHECK  
/NOREAD CHECK  
(default) Indicates whether the COPY command is to read a specific input file(s) twice to verify that all records have been correctly read.
- /REPLACE  
/NOREPLACE  
(default) Requests that, if a file already exists with the same file specification as that entered for the output file, the existing file be deleted and replaced with the contents of the input file(s).
- By default, the COPY command creates a new version of a file if the file already exists.
- /TRUNCATE  
/NOTRUNCATE  
(default) Indicates whether the command is to copy all blocks allocated to the file or only those blocks that contain data.
- When copying multiple input files into one output file, you can save space by specifying /TRUNCATE.
- If you do not specify /TRUNCATE, all of the blocks allocated to the specific file(s) are copied to the output file(s).

## MCR COMMANDS

**/VOLUME=n** Requests that the COPY command place the entire output file on the specified relative volume of a multivolume set.

If the /VOLUME keyword is not specified, the file is placed in an arbitrary position within the multivolume set.

**/WRITE CHECK**  
**/NOWRITE CHECK**  
(default) Indicates whether the COPY command is to read the output file after it has been written to. This verifies that the file has been successfully copied and that the output file can be read without error.

### Examples:

> COPY [DAVIS]JOSEPH.OBJ [HICKS]EDWARD.OBJ

This command copies the file JOSEPH.OBJ from directory [DAVIS] to directory [HICKS] on the default device and names the file EDWARD.OBJ.

> COPY \*.FOR ALLSRC.FOR

This command concatenates the highest versions of all FORTRAN source files in the default directory into a file named ALLSRC.FOR.

> COPY [JONES]ABC.DAT \*

This command copies the file ABC.DAT from the directory [JONES] to the current default directory, preserving the name as ABC.DAT.

### Notes:

1. RSX-11M MCR does not support the COPY command; however, under RSX-11M, you can use PIP instead of COPY.
2. When you use wild card characters or /CONCATENATE to concatenate Files-11 Structure Level 1 files, the input files are copied to the output file in random order. When you concatenate Files-11 Structure Level 2 files, the COPY command copies files in alphanumeric order. When wild card characters are used in the version field, files are copied in descending order by version number; that is, files with the same name are copied starting with the highest version number.
3. The VAX/VMS Command Language User's Guide provides additional examples of the use of the COPY command and further information on wild card characters.

## DEALLOCATE

### 4.8 DEALLOCATE

The DEALLOCATE command releases a previously allocated device. The system automatically deallocates any allocated devices when the owner logs off.

**Format:**

```
DEALLOCATE [keyword] [device-name[:]]
```

keyword                    /ALL

device-name[:]            Specifies the physical name or logical name assigned to the device to be deallocated. If you omit the controller designator and unit number of a physical device name, they default to controller A and unit 0, respectively.

**Command Keyword:**

/ALL                      Requests that all devices currently allocated to your process be deallocated. If you specify /ALL, you cannot specify a device name.

**Example:**

```
> DEA MT0
```

This command deallocates the physical device whose name is the equivalence for the logical name MT0.

**Note:**

1. The compatible form of the DEALLOCATE command supported by both RSX-11M and VAX/VMS MCR is:

```
DEA device-name:
```

## DEBUG

## 4.9 DEBUG

The DEBUG command invokes a debugger after you have interrupted a program's execution by pressing CTRL/C or CTRL/Y. If the image is native, the VAX-11 Symbolic Debugger prompts. If it is an RSX-11M task image that was built with ODT, ODT prints a T-bit exception message.

## Format:

```
DEBUG
```

## Examples:

```
> RUN COPLEY/NODEBUG
^Y
> DEBUG
DBG>
```

In this example, pressing CTRL/Y interrupts execution of a VAX-11 native mode image and causes MCR to prompt. Typing the DEBUG command at this point causes the VAX-11 debugger to prompt.

```
> RUN PEALE/NODEBUG
^Y
> DEBUG
-
```

In this example, pressing CTRL/Y interrupts execution of an RSX-11M task image and causes MCR to prompt. Typing the DEBUG command at this point causes ODT to prompt.

## Notes:

1. If no image is currently executing, the DEBUG command performs no operation.
2. If the interrupted program was not linked with the appropriate debugging aid (VAX-11 Symbolic Debugger for native mode images or ODT for RSX-11M task images), the DEBUG command causes a software exception condition.

If the image is a native mode image that has not declared a condition handler, or if it is an RSX-11M task image that has not specified a T-bit SST routine address, this exception may cause termination of the image.

3. For details on the use of the debugger for native mode images, see the VAX-11 Symbolic Debugger Reference Manual.
4. For details on the use of ODT for RSX-11 task images, see the IAS/RSX-11 ODT Reference Manual.
5. For details on condition handling and exception conditions, see the VAX/VMS System Services Reference Manual.

## DELETE

### 4.10 DELETE

The DELETE command deletes files, deletes entries from a print queue or a batch queue, and stops the printing of a file.

You specify /ENTRY to delete a batch or print queue entry. You can specify any of the remaining keywords to control the deletion of files. The /CREATED, /MODIFIED, and /EXPIRED keywords determine whether files are to be deleted based on their creation date, most recently modified date, or planned expiration date. The /BEFORE and /SINCE keywords are used in conjunction with the selection date keywords to establish the time span to be considered by the DELETE command.

#### Format:

```
DELETE [keywords] file-spec[,...]
```

|          |                                                                                                      |
|----------|------------------------------------------------------------------------------------------------------|
| keywords | /BEFORE[=(time)]<br>/[NO]CONFIRM<br>/CREATED<br>/EXPIRED<br>/[NO]LOG<br>/MODIFIED<br>/SINCE[=(time)] |
|----------|------------------------------------------------------------------------------------------------------|

|           |                                                                                                                                                    |
|-----------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| file-spec | Specifies the name of each file to be deleted. If you specify more than one file specification, separate each with a comma (,) or a plus sign (+). |
|-----------|----------------------------------------------------------------------------------------------------------------------------------------------------|

The first file specification must contain a file name, file type, and version number. You can specify these fields as wild card characters. Subsequent file specifications can omit any of these fields except the version numbers; the first file specification provides defaults for additional file specifications.

Before you use negative version numbers, make sure you understand the differences between RSX-11M and VAX/VMS negative version numbers, which are described in Section 2.3.3.

#### Command Keywords:

|                     |                                                                                                                                                                                                                                                                                                                                    |
|---------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| /BEFORE[=(time)]    | Chooses only those specific files that have a selection date before the specified date. The selection date can be specified as an absolute date and time, a delta time, a combination absolute and delta time, TODAY, or YESTERDAY. Section 2.7.6 explains how to specify an absolute time or combination absolute and delta time. |
| /BEFORE[=TODAY]     |                                                                                                                                                                                                                                                                                                                                    |
| /BEFORE[=YESTERDAY] |                                                                                                                                                                                                                                                                                                                                    |

If you do not specify /BEFORE, the command deletes all specified files regardless of date, unless /SINCE is specified.

## MCR COMMANDS

If you specify /BEFORE but do not supply a selection date, the DELETE command uses TODAY by default; that is, the command deletes all files created or modified yesterday or before.

/CONFIRM  
/NOCONFIRM (default)

Requests the command to display the name of each file before it is deleted and wait for you to type a confirmation. If you type a Y followed by RETURN in response to the message, the file is deleted. Any other response causes the file to be retained.

/CREATED

Requests that specific file(s) be selected for deletion based on their creation date. This keyword is meaningful only when /BEFORE or /SINCE is specified.

The default selection date is both the created date and the most recently modified date.

/EXPIRED

Selects specific files according to their planned expiration date. This keyword is meaningful only when /BEFORE or /SINCE is specified.

The default selection date is both the created date and the most recently modified date.

/LOG  
/NOLOG (default)

Determines whether the file specification of each file is to be displayed as the file is deleted. If you specify /LOG, the DELETE command displays the following information:

```
ddcu:[dir]filename.type;version deleted
```

/MODIFIED

Selects specific files according to the date on which they were most recently modified. This keyword is the default and is meaningful only when /BEFORE or /SINCE is specified.

/SINCE[=(time)]  
/SINCE[=TODAY]  
/SINCE[=YESTERDAY]

Chooses only those specific files that have a selection date after the specified date. The selection date can be specified as an absolute time, a delta time, a combination absolute and delta time, TODAY, or YESTERDAY. Section 2.7.6 explains how to specify an absolute time or a combination absolute and delta time.

If you do not specify /SINCE, the command deletes the specified files regardless of date, unless /BEFORE is specified.

If you specify /SINCE but do not supply a selection date, the DELETE command uses TODAY by default; that is, the command deletes all specified files with a selection date of today.

## MCR COMMANDS

### Example:

```
> DELETE /SINCE=(11:30) INPUT.DAT;*
```

This command deletes all versions of the file INPUT.DAT created or modified after 11:30 today.

### Note:

1. RSX-11M MCR does not support this command.

## DELETE/ENTRY

## 4.11 DELETE/ENTRY

The DELETE/ENTRY command deletes one or more entries from a print or batch job queue. The /ENTRY keyword is required.

The job(s) to be deleted must have been queued by the current process or any process in the same group as the current process if you have GROUP privilege. Otherwise, WORLD or OPER privilege is required to delete a job that is not your own or in your group.

You can delete jobs that have not yet begun processing or files that are currently being processed.

When you need to delete an entry from a batch queue while the job is running, use the STOP/ENTRY command. To delete a print job that is currently printing, consider either the STOP/ABORT or the STOP/REQUEUE commands.

The /ENTRY keyword requires at least one job-number parameter to specify the job number(s) of one or more jobs to be deleted from a print or batch job queue. If you specify more than one job number, separate the numbers with commas and enclose the list in parentheses.

## Format:

```
DELETE/ENTRY=(job-number[,...]) queue-name[:]
```

queue-name[:]            Specifies the name of the queue in which the job(s) exist.

## Examples:

```
>PRINT/HOLD ALPHA.TXT
Job 110 entered on queue SYS$PRINT
.
.
.
>DELETE/ENTRY=110 SYS$PRINT
```

The PRINT command queues a copy of the file ALPHA.TXT in a hold status to defer its printing until a later time. The system displays the job number 110 and the name of the queue in which the file was entered. Later, the DELETE/ENTRY command requests that the entry be deleted from the queue SYS\$PRINT.

```
>SUBMIT/HOLD/PARAMETERS=SCANLINE DOFOR.COM
Job 203 entered on queue SYS$BATCH
>SUBMIT/AFTER=18:00 WEATHER.COM
Job 210 entered on queue SYS$BATCH
.
.
.
>DELETE/ENTRY=(203,210) SYS$BATCH
```

## MCR COMMANDS

The SUBMIT commands spool the command procedures DOFOR.COM and WEATHER.COM for processing as batch jobs. DOFOR.COM is queued in a hold status; WEATHER.COM is queued for execution after 6:00 P.M. Later, the DELETE/ENTRY command requests that both these entries be deleted from the queue SYSSBATCH.

## DEPOSIT

## 4.12 DEPOSIT

The DEPOSIT command replaces the contents of a specific location in virtual memory. After the contents are replaced, DEPOSIT displays the location and the newly deposited data as follows:

address: contents

The DEPOSIT command converts the specified value to binary before placing it in virtual memory.

If the address specified can be read, but not written, the current contents of the location are not altered, but are displayed. If the address specified cannot be read or written, asterisks (\*\*\*\*) are displayed.

When used with the EXAMINE command, the DEPOSIT command can aid in the debugging of programs interactively. It is not necessary to link with a debugger to use these commands.

## Format:

DEPOSIT [keywords] location=data [,...]

keywords           /ASCII  
                  /BYTE  
                  /DECIMAL  
                  /HEXADECIMAL  
                  /LONGWORD  
                  /OCTAL  
                  /WORD

location           The virtual address of a location whose contents are to be changed. The specified location must be within the virtual address space of the image currently running in the process, and it must be read/write for user access mode.

You can specify the location as a symbolic name that has been assigned a value equivalent to a virtual memory location. The arithmetic operators +, -, \*, and / can be used to express the location. The value of a symbol used in an expression always is interpreted in the radix used to define it. Any numeric value used in the expression is interpreted according to the current default radix of the command. Refer to the examples that follow.

The DEPOSIT and EXAMINE commands set a pointer to the byte following the last byte modified. You can refer to this pointer as "." for the first location in a subsequent EXAMINE command or as the deposit location in a subsequent DEPOSIT command.

## MCR COMMANDS

data Data to be deposited into the specified location. If you specify a list of data, the data is deposited into consecutive locations beginning at the specified location. By default, data is assumed to be in hexadecimal format.

The DEPOSIT command converts the specified data to binary and writes it into the specified virtual memory location.

### Command Keywords:

- /ASCII** Indicates that the data specified is in ASCII format.
- When ASCII is specified or is the default, hexadecimal is the default radix for numeric literals. The location specified is interpreted as a hexadecimal address.
- /BYTE** Specifies that data is to be deposited one byte at a time. If the data specified is longer than a byte, an error message is displayed.
- /DECIMAL** Indicates that the default radix for numeric literals and displayed values is decimal.
- /HEXADECIMAL** Indicates that the default radix for numeric literals and displayed values is hexadecimal. This is the initial default of the command; refer to the Notes that follow.
- /LONGWORD** Specifies that data is to be deposited one longword at a time. This is the initial default of the command; refer to the Notes that follow.
- /OCTAL** Indicates that the default radix for numeric literals and displayed values is octal.
- /WORD** Specifies that data is to be deposited one word at a time.

### Examples:

```
> RUN RSXPROG
 .
 .
 ^Y
> EXAMINE/OCT/WORD 2332
000004DA: 007402
> DEPOSIT . = 777
000004DA: 000777
> CONTINUE
```

Pressing CTRL/Y interrupts execution of the RSX-11M task image RSXPROG. The EXAMINE command establishes the radix as octal and the length unit as a word and displays the content of location 4DA (hexadecimal). The DEPOSIT command then deposits the octal value 777 in location 4DA.

## MCR COMMANDS

```
> .SETN BASE %X200
> RUN XYZ
.
.
^y
> EXAMINE BASE:BASE+20
00000200: 00037C02 0003CFF0 6D756E20 2C726562 646E6520 74697720 39392068
0000021C: 0F13CBA1 0C07040A
> DEPOSIT BASE+16=0723C1D0
> CONTINUE
```

In the sequence above, the .SETN directive, typed interactively at the terminal, defines the symbol BASE as being equal to 200 (hexadecimal). Refer to Chapter 5 for a description of numeric literals under VAX/VMS. The EXAMINE command displays the contents of locations 200 (hexadecimal) through 220 (hexadecimal). The DEPOSIT command deposits data in location 216 (hexadecimal).

In the example above, hexadecimal is the default radix; therefore, data is deposited at location 216 (hexadecimal). If the default were octal, the data would be deposited at the location derived from 200 (hexadecimal) plus 16 (octal).

### Notes:

1. The initial default radix for the DEPOSIT command is hexadecimal. The default applies to both the interpretation of numeric literals in the command line and to the data. If you use a radix keyword to modify the command, that radix becomes the default for subsequent DEPOSIT and EXAMINE commands.
2. The initial default length unit for the DEPOSIT command is a longword. The EXAMINE command displays data one longword at a time with blanks between longwords. If you use a /BYTE or /WORD keyword to modify the command, that unit becomes the default for subsequent DEPOSIT and EXAMINE commands.
3. RSX-11M task images start at virtual address 0 under VAX/VMS; this is the same starting address that they have under RSX-11M.
4. RSX-11M MCR does not support the DEPOSIT command.

## DIRECTORY

### 4.13 DIRECTORY

The DIRECTORY command provides a list of files or information about a file or group of files.

The output of the DIRECTORY command depends on certain formatting keywords and their defaults. These keywords are: /COLUMNS, /DATE, /FULL, /OWNER, /PROTECTION, and /SIZE. However, Files-11 Structure Level 2 files usually are listed in alphabetical order, with the highest-numbered versions first. Files-11 Structure Level 1 files are listed randomly. The page width is adjusted automatically to the number of columns requested.

In studying the keywords and the capabilities they offer, watch for keywords that work together as well as keywords that override other keywords. For example, if you want the full format, you cannot expect that much information in one column. Thus, if you specify both /COLUMNS and /FULL, the number of columns you request is ignored.

#### Format:

```
DIRECTORY [keywords[,...]] [file-spec[,...]]
```

|          |                                                                                                                                                                                                                                                                                                               |
|----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| keywords | /BEFORE[=time]<br>/BRIEF<br>/COLUMNS=n<br>/CREATED<br>/[NO]DATE[=option]<br>/EXCLUDE=(file-spec[,...])<br>/EXPIRED<br>/FULL<br>/[NO]HEADING<br>/MODIFIED<br>/OUTPUT[=file-spec]<br>/[NO]OWNER<br>/PRINTER<br>/[NO]PROTECTION<br>/SINCE[=time]<br>/[NO]SIZE[=option]<br>/TOTAL<br>/[NO]TRAILING<br>/VERSIONS=n |
|----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

|                 |                                                                                                                                  |
|-----------------|----------------------------------------------------------------------------------------------------------------------------------|
| file-spec[,...] | Specifies one or more files to be listed. The syntax of a file specification determines what file(s) will be listed, as follows: |
|-----------------|----------------------------------------------------------------------------------------------------------------------------------|

- If you do not enter a file specification, the DIRECTORY command lists all versions of the files in your current default directory.
- If you specify only a device name, the DIRECTORY command uses your default directory specification.

## MCR COMMANDS

- Whenever the file specification does not include a file name and file type, all versions of all files in the specified directory are listed.
- If a file specification contains a file name and/or file type and no version number, the DIRECTORY command lists all versions.
- If a file specification contains only a file name, the DIRECTORY command assumes all file types and versions.

If you specify more than one file, separate the file specifications with either commas (,) or plus signs (+). You can use wild card characters in the directory specification, file name, file type, or version number fields of a file specification to list all files that satisfy the components you specify. See the chapter on file specifications in the VAX/VMS Command Language User's Guide for a full description of wild card characters.

### Command Keywords:

`/BEFORE[=time]`

Specifies that only those files dated earlier than a particular time be printed. You can specify one of the following values:

`absolute-time` An absolute date and time. Observe the syntax rules for date and time values specified in Section 2.7.6.

`combination of absolute and delta times` A combination of an absolute time and a delta time. See Section 2.7.6.

`TODAY` The absolute date and time representing the current day, month, and year, at 00:00:00.0 o'clock.

`TOMORROW` The absolute date and time representing 24 hours after 00:00:00.0 o'clock today.

`YESTERDAY` The absolute date and time representing 24 hours before 00:00:00.0 o'clock today.

This keyword is normally used in conjunction with one of the following keywords: `/CREATED`, `/EXPIRED`, or `/MODIFIED`. If you omit the `/BEFORE` keyword, you obtain all the files created, regardless of date. However, if you specify `/BEFORE` without a date or time, the default provides the files created prior to today.

## MCR COMMANDS

**/BRIEF (default)** Includes only the file name, type, and version number of each file to be listed. The default output format is /BRIEF. However, the /BRIEF keyword is overridden, whether specified explicitly or by default, whenever any of the following formatting keywords are specified in the command: /SIZE, /DATE, /OWNER, /PROTECTION, or /FULL.

The brief format lists the Files-11 Structure Level 2 files in alphabetical order from left to right on each line, in descending version number order. Files-11 Structure Level 1 files are listed randomly.

**/COLUMNS=n** Lists the files using the specified number of columns on each line of the display. This keyword is used in conjunction with the /BRIEF keyword (either explicitly or by default). By default, the number of columns in the brief format is four; however, you may request the brief format with as many columns as you desire. When other formatting keywords are specified in the command, they override the /COLUMNS keyword.

**/CREATED (default)** Selects the files according to their date of creation. This keyword is relevant only when used with the /BEFORE or /SINCE keywords, and should not be used with the /EXPIRED or /MODIFIED keywords. By default, when files are selected according to a particular date and time, the creation date is always used.

**/DATE[=option]**  
**/NODATE (default)** Includes the creation, expiration, or date last written for each file listed. If you omit this keyword, the default is /NODATE. However, if you specify /DATE without an option, the creation date is provided.

You may specify one of the following options with the /DATE keyword:

|          |                                                                                    |
|----------|------------------------------------------------------------------------------------|
| ALL      | Lists all three file dates in the order, left to right, CREATED, MODIFIED, EXPIRED |
| BACKUP   | Lists the date of the last back-up with each file                                  |
| CREATED  | Lists the creation date with each file                                             |
| EXPIRED  | Lists the expiration date with each file                                           |
| MODIFIED | Lists the last date the file was written                                           |

## MCR COMMANDS

**/EXCLUDE=(file-spec[,...])**

Excludes the listed file specification(s) from the directory search. You may use wild card characters, in file specification(s) as described in the VAX/VMS Command Language User's Guide. At least one file specification is required for this keyword, but the file specification must not include a device or directory specification. Separate multiple file specifications by commas, and enclose the list in parentheses.

**/EXPIRED**

Selects files according to the planned expiration date for each file. This keyword is relevant only with the **/BEFORE** or **/SINCE** keywords, and should not be used with the **/CREATED** or **/MODIFIED** keywords.

**/FULL**

Lists the following items for each file:

- File name
- File type
- Version number
- Number of blocks used
- Number of blocks allocated
- Date of creation
- Date of last back-up
- Date last modified
- Date of expiration
- File owner's UIC
- File protection
- File identification number (FID)
- File organization
- Other file attributes
- Record attributes
- Record format

You can find descriptions of these items in the Introduction to VAX-11 Record Management Services. Refer to Figure 4-1 at the end of this section for an example of this keyword used with the Directory command.

The **/FULL** keyword overrides the default brief listing format.

**/HEADING (default)**  
**/NOHEADING**

Controls whether heading lines consisting of a device description and directory specification are printed. The default output format provides this heading.

## MCR COMMANDS

- /MODIFIED** Selects files according to the last date the file was modified. This keyword is relevant only with the **/BEFORE** or **/SINCE** keywords, and should not be used with the **/CREATED** or **/EXPIRED** keywords.
- /OUTPUT[=file-spec]** Requests that the **DIRECTORY** command output be written to the file specified rather than to the current **SY\$OUTPUT** device. If you specify the **/OUTPUT** keyword without a file specification, the output is directed to **SY\$OUTPUT**. If you omit the file type in the file specification, the default file type is **LIS**.
- Wild card characters are not allowed in the file specification.
- /OWNER**  
**/NOOWNER (default)** Controls whether the owner of the file is listed. By default, the owner is not listed.
- /PRINTER** Queues the command output for printing under the name given by the **/OUTPUT** keyword. If you specify **/PRINTER** without the **/OUTPUT** keyword, the output is directed to a file named **DIRECTORY.LIS**, which is spooled for printing automatically and then deleted.
- /PROTECTION**  
**/NOPROTECTION (default)** Controls whether the file protection for each file is listed. The default is **/NOPROTECTION**, which does not list the file protection.
- /SINCE[=time]** Specifies that only those files dated after a specified time be printed. You can specify one of the following values:
- |                                                |                                                                                                          |
|------------------------------------------------|----------------------------------------------------------------------------------------------------------|
| <b>absolute-time</b>                           | An absolute date and time. Observe the syntax rules for date and time values specified in Section 2.7.6. |
| <b>combination of absolute and delta times</b> | A combination of an absolute time and a delta time. See Section 2.7.6.                                   |
| <b>TODAY</b>                                   | The absolute date and time representing the current day, month, and year, at 00:00:00.0 o'clock.         |
| <b>TOMORROW</b>                                | The absolute date and time representing 24 hours after 00:00:00.0 o'clock today.                         |
| <b>YESTERDAY</b>                               | The absolute date and time representing 24 hours before 00:00:00.0 o'clock today.                        |
- This keyword is normally used in conjunction with one of the following keywords: **/CREATED**, **/EXPIRED**, or **/MODIFIED**. If you omit the **/SINCE** keyword, you will obtain all the files created, regardless of date. However, if you specify **/SINCE** without a time or date, you will obtain all files created since today began.

## MCR COMMANDS

**/SIZE[=option]**  
**/NOSIZE (default)** Provides the file size in blocks used and/or allocated for each file listed, according to the option you specify. If you omit this keyword, the default is /NOSIZE. However, if you specify only /SIZE without an option, the listing provides the file size in blocks used, by default. The options you can specify are:

|                   |                                                       |
|-------------------|-------------------------------------------------------|
| <b>ALL</b>        | Lists both the file size in blocks used and allocated |
| <b>ALLOCATION</b> | Lists the file size in blocks allocated               |
| <b>USED</b>       | Lists the file size in blocks used                    |

**/TOTAL** Inhibits the listing of all individual file information and prints only the trailing lines as described under the /TRAILING keyword.

By default, the output format is /BRIEF, which gives this total, but also lists all the file names, file types, and their version numbers.

**/TRAILING (default)**  
**/NOTRAILING** Controls whether trailing lines that summarize the following information are output:

- Number of files listed
- Total number of blocks used per directory
- Total number of blocks allocated
- Total number of directories and total blocks used and/or allocated in all directories (only if more than one directory is listed)

By default, the output format includes most of this summary information. The /SIZE and /FULL keywords determine more precisely what summary information is included. If you omit /SIZE or /FULL, only the number of files is printed and possibly the total number of directories, if applicable. If you specify /SIZE, the number of blocks is also printed, according to the size option selected (used and/or allocated). If you specify /FULL, the number of files and the number of blocks used and allocated are all given.

**/VERSION=n** Causes the latest n versions of each of the files selected to be listed. If you omit the /VERSION keyword, by default the listing includes all versions of each file.

### Examples:

>DIRECTORY

The DIRECTORY command lists all versions of all files in the current default disk and directory in the brief format. The heading identifies the disk and directory, and the trailing line gives the total number of files.

## MCR COMMANDS

```
>DIRECTORY/VERSIONS=1/COLUMNS=1 AVERAGE.*
```

The DIRECTORY command lists only the highest versions of all files named AVERAGE in the current default directory. The format is brief, but restricted to just one column. Heading and trailing lines are provided.

```
>DIRECTORY BLOCK%%%
```

The DIRECTORY command locates all versions and types of files in the default device and directory whose names begin with the letters BLOCK and end with any three additional characters. The output format is brief, in four columns, with heading and trailing lines.

```
>DIRECTORY/TOTAL/SIZE=ALL
```

The DIRECTORY command outputs only a header and a trailing line that identifies the total number of files and the blocks used and allocated for all versions of all files in the default disk and directory.

```
>DIRECTORY/EXCLUDE=(AVER.DAT;*,AVER.EXE;*) [*...]AVER
```

The DIRECTORY command locates all versions and types of files named AVER in all directories and subdirectories on the default disk. From this list all versions of all files named AVER.DAT and AVER.EXE are excluded prior to listing and totalling.

```
>DIRECTORY-
>_/MODIFIED/SINCE=09-JUN-1982:01:30/SIZE=ALL/OWNER-
>_/PROTECTION/OUTPUT=UPDATE/PRINTER [A*]
```

The DIRECTORY command locates all files that have been modified since 1:30 A.M. on June 9, 1982 and that reside on the default disk in directories whose names begin with the letter A. It formats the output to include all versions, the size used and allocated, the date last modified, the owner, and the protection codes. The output is directed to a file named UPDATE.LIS that is spooled automatically and deleted when done.

```
> DIRECTORY/FULL *.MEM
```

```
Directory _DBB1:[SMITH] ①
```

```
CHANGES.MEM;8 ② Size: ③ 6/6 ④ Created: 17-JUN-1982 14:52 ⑦
Owner: [360,033] ⑤ Revised: 17-JUN-1982 15:18 (1) ⑧
File ID: (12244,3,0) ⑥ Expires: <None specified>
Backup: <No backup done>
```

```
⑨ File protection: System:RWED, Owner:RWED, Group:RWED, World:RE
File organization: Sequential
File attributes: Allocation=6, Extend=0
Record format: UFC, 2 byte header
Record attributes: Carriage return
```

```
⑩ Total of 1 file, 6/6 blocks.
```

Figure 4-1: Sample Output of DIRECTORY Command

## MCR COMMANDS

This command displays full directory information for the highest version of all files with a file type of MEM in the default directory.

The following notes describe the sample DIRECTORY command listed above in Figure 4-1.

- ① Disk and directory name
- ② File name, file type, and version number of each file
- ③ Number of blocks occupied by the file
- ④ Number of blocks allocated for the file
- ⑤ User identification code of the file's owner in the format:  
    [group,member]
- ⑥ File identification number (FID) in the format;  
    (file-number, file-sequence-number,relative-volume-number)
- ⑦ Date and time the file was created or last modified
- ⑧ Date and time that this version of the file was last revised,  
    and the revision number
- ⑨ Protection code associated with the file, in the format:  
    [system,owner,group,world]
- ⑩ Summary of file information, in the format:  
    Total of x files, in-use/allocated blocks.

### Notes:

1. If PIP format directories are desired, you can use PIP instead of the DIRECTORY command.
2. RSX-11M MCR does not support the DIRECTORY command.

## DMO

### 4.14 DMO

The DMO command releases a volume that was previously specified in a MOUNT command. It functions identically to the DCL command DISMOUNT.

**Format:**

```
DMO [keywords] device-name[:]
```

```
keywords /UNIT
 /([NO])UNLOAD
```

device-name[:]  
Specifies the name of the device to be dismounted. The device name can be a physical device name or a logical name.

If you specify a colon following the device name, the command interpreter strips the colon.

**Command Keywords:**

```
/UNIT Specifies, for disk volume sets, that only the
 volume on the specified device is dismounted. By
 default, the DMO command dismounts all volumes in
 a volume set.
```

```
/UNLOAD (default) Indicates that the device is to be physically
/NOUNLOAD unloaded. /UNLOAD is equivalent to pressing the
 button on the drive to unload the volume (making
 it not ready). /UNLOAD is the default action;
 specify /NOUNLOAD if you want the unit to remain
 loaded (ready).
```

**Example:**

```
> MOUNT/SHARE DBA3: PROJ123 DISK
.
.
.
> DMO DISK
```

The MOUNT command requests access to the volume labeled PROJ123 on DBA3 and places the logical name DISK in the current process's logical name table. Because the volume was requested using the /SHARE keyword, the volume may or may not already be mounted by other users.

The DMO command releases access to the volume PROJ123 for the current process and removes the name DISK from the process logical name table. The actual dismounting does not occur until all accessors have issued DMO commands.

## MCR COMMANDS

### Notes:

1. If the MOUNT command was used with the /SHARE keyword, it is not actually dismounted until all users who have mounted it have dismounted it.
2. Dismounting a volume causes the logical name assigned to the volume by the previous MOUNT command to be removed from the appropriate logical name table. The logical name is either the one specified in the MOUNT command or the default name of DISK\$volume-label or TAPE\$volume-label.
3. If a volume is mounted /SYSTEM or /GROUP, a DMO command causes the volume to be dismounted even if others are currently using it. However, dismounting the volume also results in removing the name from the system or group logical name table, which requires privilege.
4. If the device has been allocated with an ALLOCATE command, it remains allocated after the volume is dismounted.
5. If the volume has been mounted using the /NOSHARE keyword, the system automatically allocates the device at mount time and deallocates it when the volume is dismounted.
6. The /NOUNLOAD keyword provides the same functionality as the /LOCK=NOUNLOAD keyword in RSX-11M MCR for this command.
7. RSX-11M MCR supports the DMO command, but the keywords are different.

## EXAMINE

### 4.15 EXAMINE

The EXAMINE command displays the contents of virtual memory at the terminal. It is used after an RSX-11M or native image has been interrupted by CTRL/Y. If you interrupt an image while it is executing, you can examine its locations; then you can resume image execution by issuing a CONTINUE command.

The information is displayed in the following format.

```
location: content
```

Information that is inaccessible at user mode is displayed as asterisks (\*\*\*\*).

When used with the DEPOSIT command, the EXAMINE command can aid in the debugging of programs interactively. It is not necessary to link the image with a debugger to use these commands.

#### Format:

```
EXAMINE [keywords] location[:location]
```

|          |                                                                             |
|----------|-----------------------------------------------------------------------------|
| keywords | /ASCII<br>/BYTE<br>/DECIMAL<br>/HEXADECIMAL<br>/LONGWORD<br>/OCTAL<br>/WORD |
|----------|-----------------------------------------------------------------------------|

location

Specifies the address or range of addresses in virtual memory to be displayed. If you specify a range of addresses, you must separate the two addresses with a colon (:). Locations can be specified using expressions that contain the arithmetic operators +, -, \*, and /. A symbolic name also can be used to express an address. The value of a symbol used in an expression always is interpreted in the radix used to define it. Any numeric value used in the expression is interpreted according to the current default radix of the command. Refer to the examples below.

Addresses always are displayed in hexadecimal radix.

The EXAMINE and DEPOSIT commands set a pointer to the last byte examined; you can refer to this pointer as "." when specifying the first location in a subsequent EXAMINE command or as the deposit location in a subsequent DEPOSIT command.

## MCR COMMANDS

### Command Keywords:

- /ASCII** Indicates that data at the specified location is to be displayed in ASCII representation. Binary values that do not have ASCII equivalents are displayed as periods (.).
- When ASCII is specified or is the default, hexadecimal is the default radix for numeric literals.
- /BYTE** Indicates that data at the specified location is to be displayed one byte at a time.
- /DECIMAL** Indicates that the data is to be displayed in decimal format and changes the default radix for numeric literals.
- /HEXADECIMAL** Indicates that the data is to be displayed in hexadecimal format and changes the default radix for numeric literals. This is the initial default of the command; refer to the notes following the examples below.
- /LONGWORD** Indicates that data at the specified location is to be displayed one longword at a time. This is the initial default of the command; refer to the notes following the examples below.
- /OCTAL** Indicates that the data is to be displayed in octal format and changes the default radix for numeric literals.
- /WORD** Indicates that data at the specified location is to be displayed one word at a time.

### Examples:

```
> RUN RSXPROG
.
.
^Y
> EXAMINE/WORD/OCT 2677
000005BF: 007402
> CONTINUE
```

The RUN command begins execution of the RSX-11M task image RSXPROG. While the image is running, pressing CTRL/Y interrupts it. The EXAMINE command requests that the contents of the word addresses by virtual memory location 5BF (hexadecimal) be displayed in octal. After the display, the CONTINUE command causes the image to resume execution.

```
> RUN VAXPROG
.
.
^Y
> EXAMINE/LONG/ASC 1C00+50:1C00+7F
00001C50: FOOLISH CONSISTENCY IS THE HOBGOBLIN OF LITTLE M
> EXA 1C80:1C83
00001C80: INDS
```

## MCR COMMANDS

While the native image VAXPROG is executing, it is interrupted by CTRL/C. The EXAMINE command then requests that all the data in the virtual address range specified be displayed in ASCII format. Note that the addresses are expressed in hexadecimal and are interpreted as such.

```
> RUN ABC
.
.
^Y
> .SETN BASE %X200
> EXAMINE BASE:BASE+20
00000200: 143F4D00 1537FF3C ...
0000021C: 00032479
> CONTINUE
.
.
^Y
> EXAMINE/WORD/OCTAL BASE:BASE+20
00000200: 004672 467013 ...
> CONTINUE
```

In the sequence above, the .SETN directive, typed interactively at a terminal, defines the symbol BASE as being equal to 200 (hexadecimal). The first EXAMINE command displays the contents of location 200 (hexadecimal) through 220 (hexadecimal). The contents of 32 locations are displayed.

In the second EXAMINE command, the content of 16 locations are displayed. The EXAMINE command uses the current radix (octal) as the radix for the numeric value (20) in the expression.

### Notes:

1. The initial default radix for the EXAMINE command is hexadecimal. The default applies to both the interpretation of numeric literals in the command line and to the format of the data displayed. If you use a radix keyword to modify the command, that radix becomes the default for subsequent EXAMINE and DEPOSIT commands.
2. The initial default length unit for the EXAMINE command is a longword. The EXAMINE command displays data one longword at a time with blanks between longwords. If you use a /BYTE or /WORD keyword to modify the command, that unit becomes the default for subsequent EXAMINE or DEPOSIT commands.
3. RSX-11M MCR does not support the EXAMINE command. Instead of the DEPOSIT and EXAMINE commands, RSX-11M MCR uses the OPE (Open) command.
4. RSX-11M task images start at virtual address 0 under VAX/VMS; this is the same starting address that they have under RSX-11M.

## Execute Procedure (@)

## 4.16 EXECUTE PROCEDURE (@)

The Execute procedure (@) requests execution of an indirect command file or requests the command interpreter to read subsequent command input from a specified file or device.

## Format:

```
@file-spec [keyword] [p1 [p2 [...p8]]]
```

keyword                    /OUTPUT=file-spec

file-spec                 Specifies the command procedure to be executed, or the device from which input for the preceding command is to be read.

If you do not specify a file type, the MCR command interpreter uses the default file type CMD.

p1,p2,...p8               Specifies from one to eight optional parameters to pass to the indirect command file. The parameters assign numeric or character string values to the symbols, P1, P2, and so on up to P8 in the order of entry. The symbols are local to the indirect command file. Unspecified parameters are set to null strings. Separate each parameter with one or more blanks.

You can specify a numeric value for a parameter using any valid arithmetic expression. You also can specify a character string value using any alphanumeric or special characters, with the following restrictions:

- If the first parameter begins with a slash (/), you must enclose the parameter in quotation marks.
- To pass a parameter that contains embedded blanks, place the parameter in quotation marks.
- To pass a parameter that contains literal quotation marks, enclose the entire string in quotation marks and use a double set of quotation marks within the string, for example:

```
"NEVER SAY "DIE""
```

The command interpreter strips the set of quotation marks that enclose the entire string and the outermost pair of double quotation marks within the string. This results in:

```
NEVER SAY "DIE"
```

## MCR COMMANDS

Note that in each case above, the command interpreter will strip the set of quotation marks that enclose the entire string.

### Command Keyword:

`/OUTPUT=file-spec`

Requests that all output directed to the logical device `SYSS$OUTPUT` be written to the file or device specified. System responses and error messages are written to the terminal as well as to the specified file.

If you specify `/OUTPUT`, the keyword must follow the file specification of the indirect command file.

### Examples:

```
> @STDJOB.CMD PAYROL FICA
```

This command requests execution of the indirect command file `PAYROL.CMD` and provides two parameters, `PAYROL` and `FICA`. The indirect command file contains the following commands.

```
.enable substitution
RUN 'P1'
PRINT 'P1'.DAT
RUN 'P2'
PRINT 'P2'.DAT
```

When the indirect command file is executed, the two parameters are substituted into the `RUN` and `PRINT` commands. The result is that the indirect command file runs `PAYROL.EXE` and `FICA.EXE` and prints the files `PAYROL.DAT` and `FICA.DAT`.

### Notes:

1. You can cause symbol substitution by enclosing a symbol in single quotation marks. Refer to Chapter 5 for information on symbol substitution and using indirect command files.
2. When an indirect command file terminates, any data files that remain open are closed by the MCR command interpreter.

## EXIT

## 4.17 EXIT

The EXIT command terminates the processing of the current indirect command file. If the indirect file was executing within another indirect command file, control returns to the outer file.

The EXIT command is used primarily to terminate execution of the current indirect file and also maintain the status as it was prior to execution of the EXIT command.

## Format:

EXIT [status-code]

status-code Defines a value for the symbol \$STATUS, which is used as a return code to be tested by the next higher command level.

If you do not specify a status code, the current value of the symbol \$STATUS is not changed. Control returns to the outer level with the status of the most recently executed command or program.

The command interpreter uses numeric values returned by commands or programs to locate and display error messages. Each system message has a unique value associated with it. If you specify an even numeric value as a status code, either a warning or a severe error message will be displayed.

If you specify an odd value for a status code, either an information message or no message will be displayed.

## Example:

```
>@LEVEL1
```

This command requests execution of the indirect command file LEVEL1.CMD. It, in turn, contains a command requesting execution of the indirect command file LEVEL2.CMD. LEVEL2.CMD contains the following sequence, which causes a return to LEVEL1.CMD.

```
ON WARNING .GOTO ABEND
:
:
:
RUN LAST
EXIT
.ABEND: EXIT 4
```

The LEVEL2 command file contains an ON command that causes a branch to label ABEND if any error occurs. An error results in an exit from LEVEL2 with <EXSTAT> having a value of 4. If no error occurs, LEVEL2 exits with the status value supplied by LAST.

## MCR COMMANDS

When LEVEL1 regains control, it tests the value of \$STATUS using the following command.

```
.IF <EXSTAT> EQ 4 EXIT.
```

If <EXSTAT> is equal to 4, that is, if an error occurred in the LEVEL2 command file, LEVEL1 exits; otherwise, execution of LEVEL2 commands continues.

### Note:

1. RSX-11M MCR does not support the EXIT command. However, the RSX-11M indirect command file processor provides similar functions with the .EXIT directive.

## INITIALIZE

## 4.18 INITIALIZE

The INITIALIZE command formats and writes a label on a mass storage volume. The default format for disk volumes created using the MCR command INITIALIZE is Files-11 Structure Level 1. The INITIALIZE command also can initialize Files-11 Structure Level 2 volumes. The default format for magnetic tape volumes is the ANSI standard for tape labels, Level III.

You do not need any special privileges to initialize a blank disk or tape volume. If a volume has previously been written, however, your UIC must match the owner UIC on the volume, or you must have the user privilege to override volume protection. In the case of a tape that has been written, you can initialize the volume if you are allowed write access to it.

## Format:

```
INITIALIZE [keywords] device-name: volume-label
```

```
keywords /ACCESSED=n (D)
 /BADBLOCKS=(list[,...]) (D)
 /CLUSTER SIZE=n (D)
 /DATA CHECK[(option[,...])] (D)
 /DENSITY=n (T)
 /DIRECTORIES=n (D)
 /EXTENSION=n (D)
 /FILE PROTECTION=code (D)
 /GROUP (D)
 /HEADERS=n (D)
 /INDEX=position (D)
 /LABEL=VOLUME ACCESSIBILITY:"character" (T)
 /MAXIMUM FILES=n (D)
 /OVERRIDE=(option[,...]) (T)
 /OWNER UIC=uic
 /PROTECTION=code
 /[NO]SHARE (D)
 /STRUCTURE=level (D)
 /SYSTEM (D)
 /USER NAME=string (D)
 /[NO]VERIFIED (D)
 /WINDOWS=n (D)
```

D = applicable to disk only.

T = applicable to tape only.

Keyword descriptions are categorized accordingly below.

## device-name:

Specifies the name of the device on which the volume to be initialized is physically mounted. The device does not have to be allocated first; however, it is the recommended practice.

## MCR COMMANDS

**volume-label** Specifies the label to be written on the volume. For a disk volume, you can specify a maximum of 12 alphanumeric characters; for a tape volume, you can specify a maximum of 6 alphanumeric characters.

### Keywords Applicable to Disk and Tape:

**/OWNER\_UIC=[g,m]** Specifies the user identification code (UIC) to be assigned ownership of the volume and of system files on the volume. The group (g) and member (m) fields of the UIC can have a value in the range 0 through 377 (octal).

The brackets are required.

If you do not specify **/OWNER\_UIC**, your current UIC is assigned ownership of the volume.

**/PROTECTION=code** Specifies the protection to be applied to the volume. The protection determines which users can read files, write files, create directories, and delete files on the volume.

Specify the protection code according to the standard syntax rules described in Section 2.7.5. If you do not specify **/PROTECTION**, all categories of users are allowed all types of access. If you omit a category of user when specifying protection, that category is denied all access.

The system applies only read and write access restrictions for magnetic tapes; create and delete are meaningless. In addition, both the system and the owner are given read and write access regardless of what you specify in the protection code.

### Keywords Applicable Only to Disk:

**/ACCESSED=n** Specifies the number of directories to be maintained in system space for ready access. The maximum value of n is 255. Operator (OPER) privilege is required to use the **/ACCESSED** keyword.

If you do not specify **/ACCESSED**, the **INITIALIZE** command uses a value of 3 by default.

**/BADBLOCKS=(list[,...])** Specifies those areas on the volume that are faulty. The **INITIALIZE** command marks the areas as allocated so that no data is written in them.

You can specify one or more faulty areas using either or both of the formats shown below. If you specify more than one area, separate specifications with a comma and enclose the list in parentheses.

## MCR COMMANDS

`lbn[:count]` Specifies a logical block number on the disk volume and, optionally, a count of logical blocks beginning with the logical block specified, to be marked allocated.

`sector.track.cyl[:count]` Specifies a specific sector, track, and cylinder on the disk volume and, optionally, a count of blocks beginning with the first block specified to be marked allocated.

All media supplied by DIGITAL and supported on VAX/VMS, except floppy disks, TU58 cartridges, and RP04/5/6 disk packs are factory formatted and contain bad block data. The Bad Block Locator (BAD) Utility or the diagnostic formatter ESRAC may be used to refresh the bad block data or construct it for the media exceptions above. The `/BADBLOCKS` keyword is necessary only to enter bad blocks that are not identified in the bad block data for that volume.

For information about BAD, see the VAX-11 Utilities Reference Manual.

`/CLUSTER_SIZE=n`

Defines the minimum allocation unit in blocks. The maximum size that you can specify for a volume is 1/100 the size of the volume. The minimum size that you can specify is calculated by using the formula:

$$(\text{disk size}) / (255 * 4096)$$

The default cluster size for a Files-11 Structure Level 2 disk depends on the disk capacity. For disks that are 50,000 blocks or larger, the default cluster size is 3. Disks that are smaller than 50,000 blocks have a default value of 1.

Files-11 Structure Level 1 disks always have a cluster size value of 1.

`/DATA_CHECK[=(options,...)]`

`/NODATA_CHECK`  
(default)

Defines a default for data check operations following all read and/or write operations on the volume. You can specify either or both of the following options:

`READ` Performs data checks after all read operations

`WRITE` Performs data checks after all write operations

## MCR COMMANDS

If you specify /DATA\_CHECK without specifying an option, the system assumes /DATA\_CHECK=WRITE. By default, the system performs no data checking. You can override the checking you specify at initialization when you issue a Mount command for the volume.

You cannot specify /DATA\_CHECK for structure level 1 volumes.

**/DIRECTORIES=n** Specifies the number of entries to preallocate for user directories. The maximum allowable value of n is 16000. If you do not specify /DIRECTORIES, the INITIALIZE command uses a value of 16 by default.

**/EXTENSION=n** Specifies the number of blocks to use as a default extension size for all files on the volume. The default extension size is used when a file being updated increases to a size greater than its initial allocation. The maximum allowable value of n is 65535.

If you do not specify a default extension size, the INITIALIZE command uses a value of 5.

**/FILE\_PROTECTION=code** Defines the default file protection to be applied to all files on the volume. Specify the code according to the standard syntax rules for protection, as described in Section 2.7.5.

This default protection is not used when the volume is being used on a VAX/VMS system. It is used on RSX-11M systems. VAX/VMS uses your user default file protection specified in your user authorization file.

**/GROUP** Defines a disk volume as a group volume. The owner UIC of the volume defaults to the group number of the user issuing the command and a member number of 0.

The /GROUP keyword establishes the volume protection as RWED for system, owner, and group.

**/HEADERS=n** Specifies the number of file headers to be allocated initially for the index file. By default, the INITIALIZE command allocates 16 file headers.

The minimum allowable value of n is 16. The /MAXIMUM\_FILES keyword specifies the maximum.

**/INDEX=position** Requests that the index file for the volume's directory structure be placed in a specific location on the volume. You can specify one of the following options:

**BEGINNING** Places the index file at the beginning of the volume.

**MIDDLE** Places the index file in the middle of the volume.

## MCR COMMANDS

END Places the index file at the end of the volume.

BLOCK:n Places the index file at the beginning of the logical block specified.

By default, the INITIALIZE command places the index file in the middle of the volume.

**/LABEL=VOLUME\_ACCESSIBILITY: "character"**

Specifies the character to be written in the volume accessibility field of the VAX/VMS ANSI volume label VOL1 on an ANSI tape. The character can be any valid ANSI "a" character. This set of characters includes numeric characters, uppercase alphabetic characters and any one of the following:

! " & ' ( ) \* + , - . / : ; < = > ?

The default is the space character.

If you specify any character other than a space, you must specify the /OVERRIDE=ACCESSIBILITY qualifier on the INITIALIZE and MOUNT commands in order to access the tape.

**/MAXIMUM\_FILES=n**

Restricts the maximum number of files that the volume can contain which can override the default value. The default is calculated from the volume size in blocks as follows:

$$(\text{volume-size}) / ((\text{cluster factor} + 1) * 2)$$

The maximum size that you can specify for any volume is determined by the following formula:

$$(\text{volume-size}) / (\text{cluster factor} + 1)$$

The minimum value that you can specify is 0. However, note that once a value for the maximum number of files is specified, the only way to reset it is to reinitialize the volume.

**/SHARE (default)**  
**/NOSHARE**

Controls whether a disk volume is shareable. The protection code for the volume defaults to all types of access for all categories of users. If you specify /NOSHARE the protection code defaults to no access for group or world.

**/STRUCTURE=level**

Indicates the structure level of the volume. If you do not specify /STRUCTURE=2, the INITIALIZE command initializes the volume as a Files-11 Structure Level 1 volume, by default.

If you specify /STRUCTURE=1, you cannot specify /CLUSTER\_SIZE or DATA\_CHECK.

**/SYSTEM**

Defines a disk volume as a system volume. The owner UIC of the volume defaults to [1,1] and default protection allows all types of access to the volume for all users.

## MCR COMMANDS

No user privilege is required to use the /SYSTEM keyword; however, only users with system UICs can create directories on system volumes.

**/USER\_NAME=string**

Specifies a user name from one through 12 characters to be recorded on the volume. If not specified, the INITIALIZE command uses the user name under which you logged in.

**/VERIFIED(default)  
/NOVERIFIED**

Indicates whether the disk has bad block data on it. The INITIALIZE command assumes that disks contain bad block data and uses the data to mark the bad blocks as allocated. Use /NOVERIFIED to request Initialize to ignore bad block data on the disk.

**/WINDOWS=n**

Specifies the number of mapping pointers to be allocated for the file windows. When a file is opened, the file system uses the mapping pointers to access data in the file. The default number of pointers is 7; this also is the minimum allowable value of n. The maximum value of n is 80.

### Keywords Applicable Only to Tape:

**/DENSITY=n**

Specifies the density in bits per inch (bpi) at which the tape is to be written. You can specify a density of 800, 1600, or 6250.

If you do not specify /DENSITY for a blank tape, the system uses a default density of 1600 bpi. If you do not specify a density for a previously written tape, the system uses the density at which the tape was last written.

**/OVERRIDE=(option[,...])**

Requests that the accessibility or the expiration date specified for the tape be overridden. One of the following keywords must be specified as an option:

**ACCESSIBILITY** Indicates that the accessibility specified in nonblank header 1 and volume 1 labels of the tape is to be overridden. VAX/VMS never writes in these fields.

**EXPIRATION** Requests the INITIALIZE command to ignore the expiration date on a tape volume; the date is indicated by the expiration date of the first file on the volume.

You must be the owner of a tape volume (your UIC must match the UIC written on the volume) or you must have the user privilege VOLPRO to override volume protection to initialize a tape that has not reached its expiration date or has a nonblank accessibility field.

If you specify both keywords, separate each by a comma and enclose them in parentheses.

## MCR COMMANDS

### Examples:

```
> INITIALIZE/STRUCTURE=1 DBB2:MYVOL
```

This command initializes the disk volume on device DBB2 as a structure level 1 volume.

```
> @CMDFIL.CMD
.
.
.
INI DB2:MYVOL
.
.
.
```

In the sequence above, the indirect command file CMDFIL.CMD contains a command to initialize a volume. This command is in a format compatible with the RSX-11M MCR INITVOLUME command.

When the command is executed, VAX/VMS maps the device name from its RSX-11M format to the VAX/VMS physical device name format, in this case DBA2.

### Notes:

1. Many of the INITIALIZE command keywords are used to maximize input/output efficiency. For information on these parameters, see the VAX/VMS System Management and Operations Guide.
2. The compatible form of the INITIALIZE command between the VAX/VMS and RSX-11M MCR is:

```
INI device-name:volume-label
```

## Login Procedure

### 4.19 LOGIN PROCEDURE

VAX/VMS does not have a LOGIN command; rather, you gain the attention of the login procedure by pressing CTRL/C, CTRL/Y, or RETURN. Login prompts for your user name and password. The login procedure is equivalent to the RSX-11M MCR HELLO command in that it verifies your right to use the system.

#### Format:

```
Username: username [keywords]
Password: password
```

```
keywords /CLI=interpreter
 /[NO]COMMAND[=file-spec]
 /DISK=device-name[:]
```

**username** Is the user name stored in your user authorization file entry.

**password** Is the password stored in your authorization file entry. VAX/VMS does not display the password.

#### Command Keywords:

**/CLI=interpreter** Specifies the name of an alternate command interpreter. You can specify either the DCL or MCR command interpreter. If a command interpreter is not specified, your default command interpreter is used.

**/COMMAND=file-spec** Controls whether VAX/VMS executes your default login file or executes the specified command file. **/COMMAND** is assumed. Unless your user authorization file specifies a different login file, the MCR command language executes LOGIN.COM. DCL executes LOGIN.COM by default. Both command languages search your default directory.

**/DISK=device-name[:]** Specifies the name of a disk device to be associated with SYS\$DISK for the terminal session. If you do not specify **/DISK**, the default SYS\$DISK named in your authorization file is used.

**Example:**

```
<^Y>
Username: DUMPTY /CLI=MCR
Password:

Welcome to VAX/VMS Version 3.0
>
```

Pressing CTRL/Y gets the attention of login, which prompts for the user name. The user name is entered followed by keywords requesting the MCR command interpreter. After validating the user name, login prompts for the password.

**Notes:**

1. VAX/VMS automatically maps references to SY0 to the device associated with SYS\$DISK.
2. When you log in with the MCR command interpreter, it uses the login file specified in your user authorization file entry or searches your default directory for a file named LOGIN.CMD. If one is present, the MCR command interpreter executes it as an indirect command file.

## LOGOUT

### 4.20 LOGOUT

The LOGOUT command terminates a batch or an interactive terminal session. The system displays a termination message and performs any necessary cleanup operations, such as terminating the current image if one exists, dismounting any private volumes that remain mounted, and deallocating devices. Finally, it deletes your process and subprocesses, if any.

When you log out from an interactive terminal session, the brief form of the logout message appears. When an indirect command file (batch job) logs out, it writes the full form of the log out message to the log file.

#### Format:

```
LOGOUT
```

#### Examples:

The following example shows what appears at the terminal when you end an interactive terminal session with a LOGOUT command:

```
> LOGOUT
HUBBARD logged out at 23-JUN-1982 17:48:56.73
```

When an indirect command file logs out, the information written to a log file resembles the following example:

```
BELKNAP logged out at 24-JUN-1982 14:23:45.30

Accounting information:
Buffered I/O count 31 Peak working set size 100
Direct I/O count 9 Peak virtual size 300
Page faults 66 Mounted volumes 2
Elapsed CPU time 0 00:00:00.33 Elapsed time 0 00:00:13.27
```

#### Notes:

1. RSX-11M MCR does not support the LOGOUT command.
2. The BYE command is supported by both VAX/VMS MCR Command Language Interpreter and RSX-11M MCR.

## MOUNT

## 4.21 MOUNT

The MOUNT command makes a disk or magnetic tape volume or volume set and the data it contains available for processing by system commands or user programs.

## Format:

MOUNT [keywords] (device-name:[, ...] [(volume-label[, ...])] [logical-name[:]])

|          |                                  |
|----------|----------------------------------|
| keywords | /ACCESSED=n (D)                  |
|          | /[NO]ASSIST                      |
|          | /BIND=volume-set-name (D)        |
|          | /BLOCKSIZE=n (T)                 |
|          | /[NO]CACHE[=(option[, ...])] (D) |
|          | /COMMENT="string"                |
|          | /DATA CHECK[=(option[, ...])]    |
|          | /DENSITY=n (T)                   |
|          | /EXTENSION=n (D)                 |
|          | /FOREIGN                         |
|          | /GROUP (D)                       |
|          | /HDR3 (T)                        |
|          | /[NO]LABEL (T)                   |
|          | /[NO]MESSAGE                     |
|          | /[NO]MOUNT_VERIFICATION (D)      |
|          | /OVERRIDE=(option[, ...])        |
|          | /OWNER UIC=[g,m]                 |
|          | /PROCESSOR=option                |
|          | /PROTECTION=code                 |
|          | /[NO]QUOTA (D)                   |
|          | /RECORDSIZE=n (T)                |
|          | /[NO]SHARE (D)                   |
|          | /SYSTEM (D)                      |
|          | /UNLOCK (D)                      |
|          | /WINDOWS=n (D)                   |
|          | /[NO]WRITE                       |

D = applicable to disk only.

T = applicable to tape only.

Keyword descriptions are categorized accordingly below.

**device-name:** Specifies the physical device name or logical name of the device on which the volume is to be mounted.

When mounting a volume set, you can specify more than one device name. Separate the device names with either a comma (,) or plus sign (+). If you are mounting tape volumes, you can specify more volume labels than device names.

If you specify a colon (:) following the device name, the command strips it.

## MCR COMMANDS

**volume-label** Specifies the alphanumeric label written on the volume when it was initialized. For Files-11 Structure Levels 1 and 2, the volume label can have up to 12 characters; for tape volumes, it can have up to 6 characters.

If you specify more than one volume label, separate the labels with a comma (,) or plus sign (+). The volumes must be in the same volume set and the labels must be specified in the correct order.

The volume-label parameter is not required when you mount a volume with the /FOREIGN keyword, nor when you specify /OVERRIDE=IDENTIFICATION. To specify a logical name when you enter either of these keywords, type any alphanumeric characters in the volume label parameter position.

**logical-name[:]** Defines a 1- to 63-character alphanumeric string that is the logical name to be associated with the device(s). The logical name subsequently can be used to refer to the volume or volume set in commands and programs.

If you do not specify a logical name, the MOUNT command assigns the default logical name DISK\$volume-label for disk devices or TAPE\$volume-label for tape devices.

The MOUNT command places the logical name in the process logical name table unless /GROUP or /SYSTEM is requested. In the latter cases, the name is placed in the group or system logical name tables, respectively.

If you specify multiple devices and a logical name, the logical name applies to all devices specified.

### Keywords Applicable to Disk and Tape:

**/ASSIST (default)** Controls whether to allow operator or user intervention should a MOUNT request fail.  
**/NOASSIST**

If you specify /ASSIST or omit this keyword, the user and certain classes of operators are notified if a failure occurs during the MOUNT operation. If a failure occurs, the operator or user then has the choice of either aborting the mount, or correcting the error condition and allowing the mount to continue.

The operator ASSIST messages are sent to all operators enabled to receive DEVICE, and TAPE and DISK messages. Thus, if an operator assist is needed while mounting a disk device, a message is sent to both DEVICE and disk operators. See the DCL command REPLY (in the VAX/VMS Command Language User's Guide) for more information on enabling and disabling operator terminals.

## MCR COMMANDS

Any operator reply to a MOUNT request is written to SYS\$OUTPUT to be displayed on the user's terminal or written in a batch job log.

If no operator is enabled to receive and respond to a MOUNT request, a message is displayed informing the user of the situation. If the MOUNT request originated from a batch job, the MOUNT is aborted since there is no way to correct the error condition. See the VAX/VMS System Messages and Recovery Procedures Manual for a description of the error messages and their appropriate responses.

`/COMMENT="string"`

Includes the specified string with the operator request if the mount operation requires operator assistance. Used with the `/ASSIST` keyword. The text string is output to the log file and the current SYS\$OUTPUT device. Unless you enclose the string in quotation marks ("), characters are converted to uppercase and space and tab characters are removed.

`/DATA_CHECK[=(options,...)]`

Specifies that all read and write operations to the volume are to be followed by a write check operation. The write check compares the data on disk with that in memory to ensure that the transfer was successful. You can specify either or both of the following options.

`READ` Performs checks following all read operations

`WRITE` Performs checks following all write operations

If you specify `/DATA_CHECK` without including an option, the default of `/DATA_CHECK=WRITE` is used.

The data check operation is described in the VAX/VMS I/O User's Guide.

`/FOREIGN`

Indicates that the volume is not in the standard ANSI format used by VAX/VMS. If you do not specify `/FOREIGN`, the volume is assumed to be in Files-11 format.

If you specify `/FOREIGN`, the program that reads the volume must be able to process the labels on the volume, if any. For example, you must mount a DOS-formatted tape with the `/FOREIGN` keyword and use `FLX` to transfer it.

## MCR COMMANDS

The default protection applied to foreign volumes is read and write for system and owner. If you specify /GROUP in addition, group members are given read and write access. If you specify /SYSTEM or /SHARE, group and world are both given read and write access. Any user category that has read and write access to a foreign volume also is allowed logical I/O and physical I/O access to the volume.

To protect a foreign volume, you must specify the /PROTECTION keyword. If the volume is currently in Files-11 format, you must have override volume protection (VOLPRO) privilege to mount it as foreign, or you must be the volume owner (same UIC).

/MESSAGE (default)  
/NOMESSAGE

Controls whether MOUNT request messages are output to your current SYS\$OUTPUT device. Note that if you specify /NOMESSAGE during an operator-assisted mount, messages are not output to SYS\$OUTPUT, but the operator sees them.

/OVERRIDE=(option[,...])

Inhibits one or more of the following protection checks that the MOUNT command performs:

**ACCESSIBILITY** (For tapes only).  
Allows you to override a nonblank VOL1 or HDR1 accessibility field. You must have the user privilege to override volume protection (VOLPRO) or be the owner of the volume.

**EXPIRATION** (For tapes only).  
Allows you to write a tape that has not yet reached its expiration date. You must have the user privilege to override volume protection (VOLPRO) or your UIC must match the UIC written on the volume.

**IDENTIFICATION** Allows you to mount a volume when you do not know what the volume label is. If you specify /OVERRIDE=IDENTIFICATION, you can specify anything for the volume-label parameter or you can

## MCR COMMANDS

omit it; the MOUNT command ignores whatever you enter. The volume must be mounted /NOSHARE (either explicitly or by default).

**LOCK** Directs MOUNT not to write-lock the volume as a consequence of certain errors encountered while mounting it. Use this option when you are mounting a damaged volume to be repaired using the VAX-11 Verify Utility (under DCL). VOLPRO privilege or ownership of the volume is required to use this option.

**SETID** (For tapes that do not conform to ANSI standards). Allows you to inhibit checks of the file set identifier when you switch reels in a multivolume tape set.

If you specify more than one option, separate them with commas and enclose the list in parentheses.

If you have VOLPRO privilege and you specify /OVERRIDE=(ACCESSIBILITY, EXPIRATION) along with the /FOREIGN qualifier, the tape will not be read.

**/OWNER\_UIC=[g,m]**

Requests that the specified UIC be assigned ownership of the volume while it is mounted, thereby overriding the ownership recorded on the volume. If you are mounting a device /FOREIGN, /OWNER\_UIC requests an owner UIC other than your current UIC.

The brackets are required.

You must have the privilege to override volume protection to use the /OWNER\_UIC keyword, or you must be the owner of the volume.

**/PROCESSOR=option**

Allows you to control whether the volume has its own ACP process and which ACP image the process executes. The following options can be specified.

**UNIQUE** Creates a new process to contain a copy of the default ACP image for the specified device type or controller.

## MCR COMMANDS

**SAME:device** Requests that the same ACP process currently being used by the specified device be used.

**file-spec** Creates a new process to contain the specified ACP image (for example, a modified ACP).

Operator privilege is required to use the /PROCESSOR keyword.

**/PROTECTION=code**

Specifies the protection code to be assigned to the volume. All access to any file on the volume must pass both the volume protection and the protection applied to the individual file.

Specify the code according to the standard syntax rules for specifying protection as described in Section 2.7.5.

If you do not specify protection, it defaults to that assigned when the volume was initialized.

The protection specified by this keyword takes precedence over the protection implied by other keywords, for example, /SYSTEM and /GROUP. The only exception is /NOWRITE.

**/WRITE (default)**  
**/NOWRITE**

Control whether the volume can be written. By default, a volume is considered read/write. Specify /NOWRITE to provide read-only access to protect files.

/NOWRITE is equivalent to writelocking a drive.

### Keywords Applicable Only to Disk:

**/ACCESSED=n**

Specifies the approximate number of directories that will be in use concurrently on the volume. A value from 0 to 255 may be specified to override the default value that was specified when the volume was initialized. The value n is a decimal number.

Operator privilege is required to use /ACCESSED.

**/BIND=volume-set-name**

Creates or adds to a volume set that consists of one or more disks.

The volume-set-name specifies a 1- to 12-character name that identifies the volume set.

## MCR COMMANDS

When a volume set is created, the volumes are specified on a volume-label list. Each volume is assigned a relative volume number based on its position on the label list. The first volume specified is called the root volume of the volume set.

When adding one or more volumes to a set, the first volume label specified must be the root volume, unless the root volume already is online.

```
/CACHE=[(option[,...])]
/NOCACHE=[(option[,...])]
```

Control whether disk caching limits established at system generation time will be disabled or overridden. One or more of the following limits may be altered using the appropriate option:

[NO]EXTENT[=n] Enables or disables extent caching. You must have the operator (OPER) privilege and you must specify n, the number of entries in the extent cache. Note that NOEXTENT and EXTENT=0 both disable extent caching.

[NO]FILE\_ID[=n] Enables or disables file identification caching denoted by n, the number of entries given to be a value greater than 1. Note that both FILE\_ID=1 and NOFILE\_ID disable file identification caching. Operator (OPER) privilege is required.

LIMIT=n Specifies the maximum amount of free space in the extent cache in one thousandths of the currently available free space on the disk.

[NO]QUOTA[=n] Enables or disables quota cache. Operator (OPER) privilege is required and you must specify n, the number of entries in the cache. Usually,

n is set to the maximum number of active users expected on the disk when quotas are enabled. NOQUOTA and QUOTA=0, both disable quota file caching.

**WRITETHROUGH** Disables writeback caching, which only writes the file headers of files open for write when other files are closed. Thus, if you specify the WRITETHROUGH option, file headers are written to the disk on every file header operation.

When specifying more than one option, separate each by a comma, and enclose the list in parentheses.

/NOCACHE disables all disk caching on a volume. Specifying /NOCACHE is equivalent to specifying /CACHE=(NOEXTENT,NOFILE\_ID, NOQUOTA, WRITETHROUGH).

**/EXTENSION=n**

Specifies the number of blocks by which the file is extended. This number can be overridden by a command or program request. The minimum value of n is 0; the maximum is 65535. The value n is interpreted as a decimal number.

If you do not specify /EXTENSION, the extension value defaults to that specified when the volume was initialized.

**/GROUP**

Makes the volume available to any user with the same UIC group number as the user issuing the MOUNT command. These users are not required to issue a MOUNT command to gain access to the volume. However, volume protection is applied.

The logical name for the device is placed in the group logical name table. You must have the appropriate privilege to place a name in the group logical name table.

A volume mounted /GROUP remains mounted until it is explicitly dismounted; it is not automatically dismounted when its accessors log out of the system.

## MCR COMMANDS

If you issue a MOUNT command for a volume that is already mounted /GROUP, you must use the /SHARE keyword. Any additional keywords are ignored.

/MOUNT\_VERIFICATION (default)  
/NOMOUNT\_VERIFICATION

Controls whether the device is a candidate for mount verification. For a description of mount verification, see the VAX/VMS System Management and Operations Guide.

/QUOTA (default)  
/NOQUOTA

Controls whether disk quotas will be enforced on this disk volume. The /QUOTA keyword is the default, which enforces quotas for each user. /NOQUOTA inhibits quota checking. You must either own the volume or have the VOLPRO privilege to use this keyword. Refer to Chapter 3 in the VAX/VMS Command Language User's Guide for information concerning disk quotas.

/SHARE  
/NOSHARE (default)

Indicates whether the volume is shareable. If the volume has already been mounted /SHARE by another user, and you request that it be mounted with the /SHARE keyword, any other keywords you specify are ignored.

A volume mounted /SHARE is automatically dismounted when all of its accessors log out of the system.

By default, the MOUNT command assumes that a device is not shareable and allocates it.

If you have previously allocated the device and then specify the /SHARE keyword, the device is deallocated.

/SYSTEM

Makes the volume available to all users of the system as long as the UIC-based volume protection allows them access. These users are not required to issue a MOUNT command to gain access to the volume.

The logical name for the device is placed in the system logical name table. You must have the appropriate privilege to place a name in the system logical name table.

A volume mounted /SYSTEM remains mounted until it is explicitly dismounted; it is not automatically dismounted when its accessors log out of the system.

If you issue a MOUNT command for a volume that is already mounted /SYSTEM, you must use the /SHARE keyword. Any additional keywords are ignored.

## MCR COMMANDS

**/UNLOCK** Requests write access to the index file on the volume. The /UNLOCK keyword is allowed only if the volume is mounted /NOSHARE either explicitly or by default.

**/WINDOWS=n** Specifies the number of retrieval pointers to be used in mapping data when the volume files are opened. If /WINDOWS is not specified, the number of pointers defaults to the value specified when the volume was initialized. The value n is interpreted as a decimal number.

Operator privilege is required to use the /WINDOW keyword.

### Keywords Applicable Only to Tape:

**/BLOCKSIZE=n**  
**/NOBLOCKSIZE** Sets the default block size to a specific number of bytes per block, or requests using /NOBLOCK that each block contain one record.

By default, records are written to tape volumes in 2048-byte blocks, unless /FOREIGN and /NOLABEL are specified; then the default block size is 512 bytes.

**/DENSITY=n** Specifies the density (in bpi) at which the tape is written for either foreign or unlabeled tapes. You can specify a density of 800, 1600, or 6250. To specify a specific density, either the /FOREIGN or /NOLABEL keyword must be specified and the first operation performed on the tape must be a write.

If you specify /LABEL, or if the first operation on the tape is a read, the tape is read or written at the density at which the first records on the tape are recorded.

**/HDR3 (default)**  
**/NOHDR3**

Controls whether ANSI HDR3 labels are written on magnetic tapes. ANSI HDR3 labels are written on tapes by default. For tapes which will be used on systems which do not tolerate HDR3 labels, specify /NOHDR3.

**/LABEL (default)**  
**/NOLABEL**

Indicates whether the tape contains standard labels.

If you mount a tape specifying the /NOLABEL keyword, an end-of-file condition is returned when a tape mark is encountered when reading the tape.

## MCR COMMANDS

The default protection for unlabeled tapes is all access to the system and owner and no access to the group and world.

`/RECORDSIZE=n`

Specifies the number of bytes in each record. This keyword is normally used with the `/FOREIGN` or `/BLOCKSIZE` keywords to read and write fixed-length records on a block-structured device. In this case, the record size must be less than or equal to the default block-size. The block size can range from 20 through 65,532 with VAX-11 RMS or 18 through 65,534 without VAX-11 RMS.

Use the `/RECORDSIZE` keyword whenever you are mounting tapes without HDR2 labels (such as RTM tapes). This provides VAX-11 RMS with default values for both the maximum record size and the length of the largest record.

### Examples:

```
> ASN DBA1:=DK0:
> @RODIN
 MOUNT DK0:MYVOL
% MOUNT-I-MOUNTED, MYVOL mounted on _DBA1:
.
.
.
```

The `ASN` command defines `DBA1` as the equivalence name for the logical name `DK0`. When the indirect file `RODIN.COMD` executes, its command to mount the volume labeled `MYVOL` on `DK0` becomes, in effect, a request to mount `MYVOL` on `DBA1`.

```
> MOUNT DBA1 MYVOL DISK
% MOUNT-I-MOUNTED, MYVOL mounted on _DBA1:
```

This command mounts the volume `MYVOL` on disk device `DBA1` and assigns a logical name of `DISK`.

### Notes:

1. The form of the `MOUNT` command that is compatible between `RSX-11M` and `VAX/VMS MCR` is as follows:

```
MOU llnn:volume-label
```

`llnn` is an `RSX-11M` physical device name that has been assigned as the logical name for a `VAX/VMS` physical device unit.

`volume-label` is the volume label.

2. `VAX/VMS` and `RSX-11M MCR` do not support the same keywords for the `MOUNT` command.
3. `VAX/VMS MCR` syntax for the `MOUNT` command accepts either a colon (`:`) or a colon (`:`) and a space between the device name and the volume label.

## ON Severity-Level

### 4.22 ON SEVERITY-LEVEL

The ON severity-level statement defines the default course of action to be taken when a command or program executed within an indirect command file encounters an error condition.

Return status values in VAX/VMS include a severity-level code. The code groups status returns to indicate success, warning, error, or severe error. When a command or program is executed, the return status value is compared with the current severity level to determine subsequent action.

The ON severity-level statement is used with the SET ON command. By default, the initial state for an indirect command file is to have condition recognition disabled (SET NOON). You can enable condition recognition by including a SET ON command or an ON statement in the indirect command file.

#### Format:

```

 ON severity-level THEN command
severity-level Specifies the severity of the condition that is to
 cause the indicated action to be taken. The
 severity level is represented by one of the
 following keywords.
```

```

WARNING The specified action is to be taken
 for warning, error, and severe
 error returns.
```

```

ERROR The specified action is to be taken
 for error and severe error returns;
 the default action for warnings is
 to continue.
```

```

SEVERE_ERROR The specified action is to be taken
 only for severe errors; the
 default action for warnings and
 errors is to continue.
```

You can truncate these keywords to one or more characters.

```

command Specifies the action to be taken when errors equal
 to or greater than the specified level of error
 occur. You can specify any valid MCR command line
 following the keyword THEN.
```

If you do not include an ON statement in an indirect command file and a SET ON command has been executed, the command interpreter issues an EXIT command when errors or severe errors result from the execution of a command or program. In all other cases, command execution continues.

## MCR COMMANDS

Once the command in an ON statement has been executed as a result of an error condition, the default action, EXIT, is reinstated. It remains in effect until a new ON command is encountered.

### Example:

```
> @INDFILE
 RUN POMME
 .
 .
 ON ERROR THEN CONTINUE
 RUN SQUASH
 RUN PEAS
 .
 .
 ON SEVERE ERROR THEN .GOTO ERR
 RUN TOMATO
 RUN PEPPER

.ERR: RUN MARKET
EXIT
```

When execution of the indirect command file starts, any error more severe than a warning causes the command interpreter to issue an EXIT command by default.

Once the first ON statement is encountered, the indirect command file can continue after any type of error. It continues from warnings by default and from errors and severe errors as a result of the CONTINUE command in the ON statement.

Once the second ON statement is encountered, the indirect command file is allowed to continue from warnings and errors, but severe errors cause a branch to the portion of the indirect command file labeled ERR.

### Notes:

1. Once the clause (THEN command) of a condition statement is executed, condition recognition is set to ON ERROR THEN EXIT.
2. Any ON severity-level statement in an indirect command file overrides the effect of a previous ON statement.
3. You can control whether the command interpreter checks the status returned from commands with the SET ON command.
4. You can use the SET NOON command to temporarily disable condition recognition.
5. RSX-11M MCR does not support this form of the ON statement.

## PRINT

### 4.23 PRINT

The PRINT command queues one or more files for printing on either a system-defined printer or a device that you specify. The system considers a file or files queued by a PRINT command to be a job. It assigns a unique job identification to each job in the system.

#### Format

```
PRINT [keywords] file-spec[file-keyword[,...]][,...]
```

|          |                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| keywords | <pre> /AFTER=time /[NO]BURST (file-keyword) /CHARACTERISTICS=(c[,...]) /COPIES=n (file-keyword) /[NO]DELETE (file-keyword) /DEVICE=device-name[:] /[NO]FEED (file-keyword) /[NO]FLAG_PAGE (file-keyword) /FORMS=type /[NO]HEADER (file-keyword) /[NO]HOLD /[NO]IDENTIFY /JOB_COUNT=n /[NO]LOWERCASE /NAME=job-name /[NO]NOTIFY /PAGE_COUNT=n (file-keyword) /PRIORITY=n /QUEUE=queue-name[:] /REMOTE /[NO]SPACE (file-keyword) </pre> |
|----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

|           |                                                                                                                                                                     |
|-----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| file-spec | <p>Provides the file specification of one or more files to be printed. If you specify multiple files, separate them with either a comma (,) or a plus sign (+).</p> |
|-----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|

You can use wild card characters in place of the directory, file name, type, or version fields of the file specification.

If you omit a file type, the PRINT command uses LIS by default.

#### Command Keywords:

|             |                                                                                                                                                                                              |
|-------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| /AFTER=time | <p>Requests that the job be printed after a specific time of day. The time can be specified as an absolute time, delta time, or combination absolute and delta time (see Section 2.7.6).</p> |
|-------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

For example /AFT=17 places a job in a hold status until 5:00 P.M. If the time already has passed, the job is printed immediately.

## MCR COMMANDS

- /CHARACTERISTICS=(c[,...])** Specifies one or more characteristics desired for printing the file(s). If you specify more than one characteristic, separate each by a comma and enclose the list in parentheses. For details on specifying characteristics, see the VAX/VMS System Management and Operations Guide.
- /DEVICE=device-name[:]** Requests that the files specified be queued to a particular device. If you do not specify /DEVICE, files are queued to the current default printer. The default device is SYSS\$PRINT.
- /FORMS=type** Specifies the form type required for the files to be printed. Specify the form type using a numeric value or alphanumeric code. Codes for form types are installation defined.
- /HOLD**  
**/NOHOLD (default)** Request that the file(s) be queued but not printed. The file cannot be released for printing until you issue the SET QUEUE command to release it.
- /IDENTIFY (default)**  
**/NOIDENTIFY** Control whether the PRINT command displays a message indicating the job identification of the print job and the name of the device to which it has been queued.
- By default, the PRINT command displays this information in the following format:
- JOB jobid ENTERED ON QUEUE device-name
- /JOB\_COUNT=n** Requests that the entire job be printed n times. The value of n is a decimal number ranging from 1 through 255. If you omit this keyword, one copy of the job is printed.
- /LOWERCASE**  
**/NOLOWERCASE (default)** Indicate that the specified file(s) contains lowercase alphabetic letters and must be printed on a printer that has uppercase and lowercase capabilities.
- /NAME=job-name** Defines a 1- to 8-character alphanumeric string to identify the job. The name string is displayed in response to the SHOW QUEUE command.
- If name is not specified, the name string defaults to the file name of the first file in the job.
- /NOTIFY**  
**/NONOTIFY (default)** Controls whether a message is broadcast to any terminal at which you are logged on, notifying you when your print job has been completed or aborted.
- /PRIORITY=n** Specifies the priority of the print job. The value of n must be in the range of 0 through 31; 0 is the lowest priority, and 31 is the highest.

## MCR COMMANDS

By default, jobs are assigned the same priority as your process. You can assign lower priorities to your nonessential jobs so that jobs you want sooner will be processed first.

Privilege is required to set a priority value that is higher than your process's priority.

**/QUEUE=queue-name[:]**

Requests that the file(s) specified be queued to a particular device. The default is SYS\$PRINT.

This keyword performs the same function as /DEVICE.

**/REMOTE**

Indicates that the specified files be printed on a remote node. Because the file to be printed must exist on the remote node, the file specification must contain the name of the remote node. If this keyword is specified, the file is queued for printing to SYS\$PRINT of the node on which the file resides. See also the DECnet-VAX User's Guide.

If you specify /REMOTE, you cannot specify any other keywords.

### File Keywords:

**/BURST**

**/NOBURST (default)**

Indicate whether a burst page is to precede the file. A burst page is printed before a flag page and contains the same information; however, it is printed over the perforation between pages to make the separation of files easier.

This keyword overrides the installation's default established for printers when they are initialized for spooling. /NOBURST is the default for user-specified devices.

**/COPIES=nn**

Specifies the number of copies of the file to be printed. The value of nn is a decimal number in the range of 1 through 255. The default number of copies is 1.

If you specify /COPIES as a command keyword, each file named in the command is printed the specified number of times.

**/DELETE**

**/NODELETE (default)**

Control whether the file is to be deleted after printing.

If you specify /DELETE after the PRINT command all files specified are deleted. Files are deleted after printing by default.

Note that the protection code must allow delete access to the current UIC.

## MCR COMMANDS

|                                                                |                                                                                                                                                                                                                                                                                                                                                                                     |
|----------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>/FEED (default)</code><br><code>/NOFEED</code>           | Control whether the PRINT command automatically inserts form feeds when it prints files that do not have carriage control characters. By default, the PRINT command inserts a form feed when the forms are within four lines of the end of the form. On standard 66-line forms, a form feed occurs after printing 62 lines.                                                         |
| <code>/FLAG_PAGE</code><br><code>/NOFLAG_PAGE (default)</code> | Indicate whether a flag page is to be printed preceding the file. If you specify <code>/FLAG</code> with a file specification, a separate flag page is printed preceding the associated file. The flag page lists the file name.<br><br>If you specify <code>/FLAG</code> with the command verb, a flag page is printed for each file in the job; the flag page lists the job name. |
| <code>/HEADER</code><br><code>/NOHEADER (default)</code>       | Control whether the name of the file is printed at the top of each printed page.                                                                                                                                                                                                                                                                                                    |
| <code>/PAGE_COUNT=n</code>                                     | Specifies the number of pages of the file to be printed. <code>/PAGE_COUNT</code> can be used only as a file keyword; it cannot be used as a command keyword. If this keyword is omitted, the entire file is printed.                                                                                                                                                               |
| <code>/SPACE</code><br><code>/NOSPACE (default)</code>         | Specifies whether output is to be double-spaced. By default, output is single-spaced.                                                                                                                                                                                                                                                                                               |

### Example:

```
> PRINT/COPIES=5 THISFILE.DAT/SPACE,THATFILE/PAGE_COUNT=4
```

This command requests the printing of five copies of THISFILE and THATFILE. THISFILE is to be double-spaced. Only the first four pages of THATFILE are to be printed.

### Notes:

1. You can delete a queue entry using the DELETE command with the `/ENTRY` keyword.
2. You can control the attributes of a print queue using the SET QUEUE command.
3. RSX-11M optionally supports the PRINT Utility.

## PURGE

### 4.24 PURGE

The PURGE command deletes all but the highest numbered version or versions of a specified file or files.

#### Format:

```
PURGE [keyword] file-spec[,...]
```

#### keywords

```
/KEEP=n
/[NO]LOG
```

#### file-spec

Provides the file specification of the files to be purged. If you specify more than one file, separate each with either a comma (,) or a plus sign (+). Do not specify a version number.

You can use wild cards in place of the file name or file type fields.

If you do not provide the file name and type, the PURGE command purges all files in the directory.

#### Command Keywords:

```
/KEEP=n
```

Specifies the maximum number of versions of the specified files to be retained. If /KEEP is not specified, the command deletes all but the highest numbered version.

```
/LOG
/NOLOG (default)
```

Specifies that the names of files be displayed as they are deleted. /NOLOG is the default, thus file names are not displayed as they are purged.

#### Examples:

```
> PURGE /KEEP=2 *.DAT
```

This command requests that all files in the default directory having a file type of DAT be purged so that only the two highest numbered versions remain.

```
> PURGE [122020.MSG]
```

This command requests the purging of all files in subdirectory [122020.MSG] to one version.

## MCR COMMANDS

### Notes:

1. RSX-11M MCR does not support the PURGE command.
2. You can use the /PU switch with PIP to purge files.
3. PURGE differs from the operation of PIP/PU in that the /KEEP keyword specifies the actual number of files to be kept rather than the highest version numbers. The difference occurs only when versions of the file have been deleted. If the highest version number is 15 and version number 13 has been deleted, PURGE/KEEP=3 deletes all versions except 12, 14, and 15. Using PIP name/PU:3 keeps only versions 14 and 15, however, because PIP calculates  $15-3=12$  and keeps only version numbers greater than 12. In summary, PIP calculates and PURGE counts.

## RENAME

### 4.25 RENAME

The RENAME command changes the directory specification, file name, file type, or version number of an existing disk file or disk directory.

Format:

```
RENAME [keywords] input-file-spec [...] output-file-spec
```

|                  |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| keywords         | <pre>/[NO]CONFIRM /[NO]LOG /[NO]NEW_VERSION</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| input-file-spec  | <p>Specifies the names of one or more files whose specifications are to be changed.</p> <p>You can use wild card characters in the directory specification, file name, file type, or version number fields. By using wild card characters, all files whose specifications satisfy the fields that are specified are renamed.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| output-file-spec | <p>Provides the new file specification to be applied to the input file. The RENAME command uses the device, directory, file name, and file type of the input file specification to provide defaults for nonspecified fields in the output file.</p> <p>You can specify an asterisk (*) in place of the directory specification, file name, or version number of the output-file-spec parameter; the RENAME command uses the corresponding field in the input file specification to name the output file. Wild card characters in corresponding fields of the input and output file specification result in multiple rename operations.</p> <p>The RENAME command supplies output file version numbers according to the first description below that applies:</p> <ul style="list-style-type: none"> <li>• If the output file specification contains an explicit version number, RENAME uses that version number.</li> <li>• If the input file specification or output file specification contains an asterisk in the version number field, RENAME uses the version number of each input file to name the output file.</li> <li>• If no file currently exists with the same file name and file type as that specified for the output file, RENAME assigns a file number of 1.</li> </ul> |

## MCR COMMANDS

- If a file currently exists with the same file name and file type as that specified for the output file name, RENAME assigns the output file a version number one greater than the highest existing version, unless the /NONEW\_VERSION keyword is specified.

### Command Keywords:

/CONFIRM  
/NOCONFIRM (default)

Controls whether RENAME displays the file specification of each file before renaming and asks you to confirm whether or not you want to rename the file. If you specify /CONFIRM, you must respond to a prompt by entering a Y (YES) or a T (TRUE) before the RENAME command renames the file. If you enter anything else, such as N or NO, the file is not renamed.

/LOG  
/NOLOG (default)

Controls whether RENAME displays the file specification of each file it renames.

/NEW\_VERSION (default)  
/NONEW\_VERSION

Controls whether RENAME automatically assigns a new version number to the output file, if a file with the same name and file type already exists.

By default, RENAME automatically assigns a new version number to the file if a file with the same name and file type exists.

### Examples:

```
> RENAME AVERAGE.OBJ OLDAVER
```

The RENAME command changes the file name of the highest existing version of the file AVERAGE.OBJ to OLDAVER.OBJ. If no file named OLDAVER.OBJ exists, the new file is given a version number of 1.

```
> RENAME *.TXT;* *.OLD;*
```

This command renames all versions of all files with file types of TXT to the same name except a file type of OLD. The file names and version numbers are not changed.

### Note:

1. RSX-11M MCR does not support the RENAME command.

## RUN (Image)

### 4.26 RUN (IMAGE)

The RUN (Image) command places an image into execution in the process.

You can truncate the RUN command to a single letter, R.

#### Format:

```
RUN file-spec
```

```
keyword /[NO]DEBUG
```

```
file-spec Specifies an executable image to be executed. If
 you do not specify a file type, the RUN command
 uses the default file type of EXE.
```

No wild card characters are allowed in the file specification.

#### Command Keyword:

```
/DEBUG
/NODEBUG
```

Controls, for native VAX-11 images only, whether the image is to be run with the debugger. If the image was linked with the /DEBUG keyword and you do not want the debugger to prompt, use the /NODEBUG keyword. If the image was linked with /TRACEBACK, traceback reporting is performed when an error occurs.

If the image was not linked with the debugger, you can specify /DEBUG to request the debugger at execution time. However, if /NOTRACEBACK was specified when the image was linked, /DEBUG is invalid.

#### Examples:

```
> RUN LIBRA
```

The image LIBRA.EXE starts executing in the process.

```
>MAC TEST, TEST/-SP=TEST
>TKB TEST/DA, TEST/-SP=TEST
>RUN TEST
ODT:XXXXX
```

An RSX-11M MACRO-11 image named TEST is assembled, built, and run with ODT.

#### Note:

1. The form of the RUN command that is compatible between RSX-11M and VAX/VMS MCR is:

```
RUN file-spec
```

## RUN (Process)

### 4.27 RUN (PROCESS)

The RUN command requests execution of a native or RSX-11M image. You can request an image for execution in your process (the requesting process), or you can use the RUN command to create a subprocess or detached process to execute the image. By default, the image executes in the requesting process.

When you specify keywords with the RUN command, the command creates a separate process to execute a specific image. The command displays the created process's identification on SYS\$OUTPUT. By default, it creates a subprocess with the same UIC, privileges, and priority as the requesting process and deducts resource quotas from the requesting process to assign them to the subprocess.

The /UIC keyword requests the RUN command to create a detached process; you must have the appropriate privilege to create a detached process.

Additional keywords allow you to schedule the process, describe its resources and privileges, and indicate whether the process is to hibernate.

#### NOTE

The notes at the end of this command description contain important information. You should read them if you intend to use any RUN command keywords.

#### Format:

RUN [keywords] file-spec

|          |                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| keywords | /[NO]ACCOUNTING<br>/AST LIMIT=quota<br>/[NO]AUTHORIZE<br>/BUFFER LIMIT=quota<br>/DELAY=delta-time<br>/ERROR=file-spec<br>/FILE LIMIT=quota<br>/INPUT=file-spec<br>/INTERVAL=delta-time<br>/IO BUFFERED=quota<br>/IO DIRECT=quota<br>/MAILBOX=unit<br>/MAXIMUM WORKING SET=quota<br>/OUTPUT=file-spec<br>/PAGE FILE=quota<br>/PRIORITY=n<br>/PRIVILEGES=privilege[,...]<br>/PROCESS NAME=process-name<br>/QUEUE LIMIT=quota<br>/[NO]RESOURCE_WAIT |
|----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

## MCR COMMANDS

/SCHEDULE=absolute-time  
/[NO]SERVICE FAILURE  
/SUBPROCESS\_LIMIT=quota  
/[NO]SWAPPING  
/TIME LIMIT=limit  
/UIC=[g,m]  
/WORKING\_SET=default

file-spec Provides the file specification of the native or RSX-11M image to be executed. If the file specification does not include a file type, the RUN command uses EXE by default.

### Command Keywords:

/ACCOUNTING (default)  
/NOACCOUNTING Controls whether accounting records are to be logged for the created process. By default, all processes are logged in the system accounting file.

You must have the user privilege ACNT to disable accounting.

/AST\_LIMIT=quota Specifies the maximum number of Asynchronous System Traps (ASTs) the created process can have outstanding.

If you do not specify an AST limit quota, the default value established at system generation time is used; the minimum required for any process to execute is 2. A value of 10 is typical.

This quota is nondeductible.

/AUTHORIZE (default)  
/NOAUTHORIZE Controls, when the image to be executed is the system login image (LOGINOUT.EXE), whether login searches the user authorization file to validate a detached process.

By default, the login image checks the user authorization file whenever a detached process is created. Specify /NOAUTHORIZE to create a detached process running under the control of the command interpreter. The process permanent files specified by the /INPUT and /OUTPUT keywords are made available to the command interpreter for input and output.

The user privilege DETACH is required to create a detached process. Any nonspecified attributes of the created process default to the same as those of the current process.

/BUFFER\_LIMIT=quota Specifies in bytes the maximum amount of memory that the created process can use for buffered I/O operations or temporary mailboxes. Mailboxes are used to emulate RSX-11M send/receive directives; see the notes at the end of this section.

## MCR COMMANDS

If you do not specify a buffered I/O limit quota, the default value at system generation time is used; the minimum amount required for any process to execute is 1024 bytes. This quota is pooled; see the notes at the end of this section.

**/DELAY=delta-time**

Requests that the created process be placed in hibernation and awakened after a specified time interval has elapsed.

If you specify /INTERVAL with /DELAY, the first wake-up request is scheduled for the delay time specified and all subsequent wake-up requests are scheduled according to the specified time interval.

**/ERROR=file-spec**

Defines a 1- to 63-alphanumeric character equivalence name string for the logical device name SYS\$ERROR. The logical name and equivalence name are placed in the process logical name table for the created process.

**/FILE\_LIMIT=quota**

Specifies the maximum number of files that a process can have open at one time.

If you do not specify an open file quota for a created process, the system uses the default value established at system generation time. The minimum amount required for a process to execute is 2.

This quota is pooled; see the notes at the end of this section.

**/INPUT=file-spec**

Defines a 1- to 63-alphanumeric character equivalence name string for the logical device name SYS\$INPUT. The logical name and equivalence name are placed in the process logical name table for the created process.

**/INTERVAL=delta-time**

Requests that the created process be placed in hibernation and awakened at regularly scheduled intervals as specified by the value of delta time.

If you specify /DELAY or /SCHEDULE with /INTERVAL, the first wake-up request occurs at the delay time or scheduled time specified. All subsequent wake-up requests occur at the specified interval time. If neither /DELAY nor /SCHEDULE is specified, the first wake-up request occurs immediately.

If the image to be executed is an RSX-11M image, the /INTERVAL keyword has the effect of the /DELAY keyword. Only one wake-up occurs.

## MCR COMMANDS

`/IO_BUFFERED=quota`

Specifies the maximum number of system-buffered I/O operations the created process can have outstanding at a time.

If you do not specify a buffered I/O quota, the default value established at system generation time is used; the minimum required for any process to execute is 2. A value of 6 is typical.

This quota is not deductible; see the notes below.

`/IO_DIRECT=quota`

Specifies the maximum number of outstanding direct I/O operations permitted for the created process at a time.

If you do not specify a direct I/O quota, the default value established at system generation time is used. The minimum value of n that is required for any process is 2.

This quota is not deductible; see the notes at the end of this section.

`/MAILBOX=unit`

Specifies the unit number of a mailbox to receive a termination message when the created process is deleted.

Mailbox creation and use and process termination mailboxes are described in the VAX/VMS System Services Reference Manual.

`/MAXIMUM_WORKING_SET=quota`

Specifies the maximum size to which the image to be executed in the process can increase its working set size.

If you do not specify a working set quota, the system uses the default value established at system generation time. The minimum working set size required for a process to execute is 10 pages.

This quota is not deductible; see the notes at the end of this section.

`/OUTPUT=file-spec`

Defines a 1- to 63-alphanumeric character equivalence name string for the logical device name `SYSS$OUTPUT`. The logical name and equivalence name are placed in the process logical name table for the created process.

`/PAGE_FILE=quota`

Specifies the maximum number of pages that can be allocated in the paging file for the process.

## MCR COMMANDS

If you do not specify a paging file quota, the system uses the default value established at system generation time. The minimum number of pages required for a process to execute is one 256-page block.

This quota is pooled; see the notes at the end of this section.

**/PRIORITY=nn**

Specifies the base priority at which the created process is to be executed. VAX/VMS priorities range from 0 through 31 (decimal). Priorities 0 through 15 are normal process priorities. Priorities 16 through 31 are real-time priorities.

You must have the ALTPRI privilege to set the created process's base priority to a value higher than your process's base priority. If you specify no priority or a higher priority than allowed, the system uses your process's priority by default.

**/PRIVILEGE=privilege[,...]**

Defines privileges for the created process. The privilege list consists of one or more of the privileges listed in Appendix A. You must have the SETPRV user privilege to give a process you create any privileges that you do not have.

If you specify multiple privileges, each must be separated by a comma, and the list must be enclosed in parentheses.

As listed, any of the privileges can be preceded by NO to deny the process that privilege, for example, NOSWAP\_MODE.

Rather than specifying individual privileges, you can specify either ALLPRIV or SAME. ALLPRIV indicates that the created process is to have all privileges. SAME indicates that the created process is to have the same privileges as your process. If you specify /PRIVILEGE=NOSAME, the created process has no privileges.

If you do not specify /PRIVILEGE, the created process has the same privileges as your process; that is, the default is PRIVILEGE=SAME.

**/PROCESS=process-name**

Defines a 1- to 15-alphanumeric character name for the created process. The process name is implicitly qualified by the group number of the process's UIC; that is, process names must be unique only within the group.

If you do not specify a process name, the process has a null name, by default.

## MCR COMMANDS

`/QUEUE_LIMIT=quota`

Specifies the maximum number of timer queue entries that the created process can have outstanding at any one time. This includes timer requests and scheduled wake-up requests. Timer requests are used to emulate RSX-11M MARK TIME directives.

If you do not specify a timer queue entry quota, the system uses the default value established at system generation time. A process does not require any timer queue limit to execute.

This quota is pooled; see the notes at the end of this section.

`/RESOURCE_WAIT`  
(default)  
`/NORESOURCE_WAIT`

Enables or disables resource wait mode for the created process. By default, if a system resource is required for a process to execute a particular function and the resource is not available, the system places the process in a wait state until the resource becomes available.

If you specify `/NORESOURCE_WAIT`, the process receives an error status code when the resource is not available.

`/RESOURCE_WAIT` is the default.

`/SCHEDULE=absolute-time`

Requests that the created process be placed in hibernation and awakened at the time of day specified by the value of absolute time.

`/SERVICE_FAILURE`  
`/NOSERVICE_FAILURE`  
(default)

Enables or disables system service failure exception mode for the created process. By default, if an error occurs when a process calls a system service either directly or as a result of RSX-11M directive emulation, a status code indicating the error is returned.

If you specify `/SERVICE_FAILURE` and an error occurs during the processing of a system service request, the process receives an exception condition. Native images can declare condition handlers for such exception conditions.

`/SUBPROCESS_LIMIT=quota`

Specifies the maximum number of subprocesses that the created process is allowed to create.

If you do not specify a subprocess limit, the system uses the default value established at system generation time. A process does not require any subprocess quota to execute.

This quota is pooled; see the notes at the end of this section.

## MCR COMMANDS

`/SWAPPING (default)`  
`/NOSWAPPING`

Enables or disables swapping for the created process. By default, a process is swapped from the balance set in physical memory to allow other processes to execute. Swapping is similar in function to RSX-11M checkpointing.

You must have the PSWAPM user privilege to specify `/NOSWAPPING` for a process that you create. If you specify `/NOSWAPPING`, the process is not swapped out of the balance set when it is in a wait state.

`/TIME_LIMIT=limit`

Specifies the maximum amount of CPU time allocated to the created process, in delta time, where the resolution is to ten milliseconds. When the time expires, the process is deleted. The default value is established at system generation time. A CPU time limit of 0 indicates that CPU time is not restricted; this is a typical value.

If you restrict CPU time for a process, specify the time limit according to the rules for specifying delta time values, as given in Section 2.7.6.2.

This quota is deductible.

`/UIC=[g,m]`

Specifies that the created process is to be a detached process. The [g,m] format defines the user identification code (UIC) for the created process. The values of g and m are numeric strings which respectively represent the group and member numbers of the process. The number is in the range of 0 through 377 in octal radix.

`/WORKING_SET=default`

Specifies the default working set size for the created process; that is, the number of pages in the working set for the image to be executed.

If you do not specify a default working set size, the system uses the default value established at system generation time. The minimum number of pages required for a process to execute is 10 pages. The value specified cannot be greater than the working set quota specified by the `/MAXIMUM_WORKING_SET` keyword.

This quota is not deductible.

**Examples:**

```
> RUN RSXIMAGE.TSK
```

This command activates the RSX-11M task image RSXIMAGE in the requesting process; no subprocess or detached process is created. The system does not prompt for subsequent commands until the task image terminates.

```
> RUN VAXIMAGE /UIC=[122,20]
```

This command creates a detached process that runs under UIC [122,20] and has the same privileges and resource quotas as the requesting process.

```
> RUN FIRST.TSK /PROCESS=PAY1 /DELAY=:::10
> RUN SECOND.TSK /PROCESS=PAY2
```

The commands above create two subprocesses named PAY1 and PAY2, each of which is to execute an RSX-11M task image. PAY1 hibernates as a result of the /DELAY keyword on its RUN command; PAY2 is activated immediately. The two processes cooperate. SECOND, the image executing in PAY2, awakens PAY1 by issuing a directive.

The process requesting that the two images be run remains free to accept subsequent user commands.

Both subprocesses have the same privileges as their requesting process. Resource quotas are subtracted from the requesting process's.

**Notes:**

1. Specifying a process name is not sufficient to allow emulation of the RSX-11M functions (for example, common event flag clusters and send/receive) that require a task name in the image label block.
2. If you run an RSX-11M task image that has a task name starting with an alphabetic character in the image label block and specify /PROCESS, the task name becomes the process name; that is, it overrides the name specified by /PROCESS.
3. Mailboxes are virtual devices that can be used by both RSX-11M and native images. VAX/VMS uses mailboxes to emulate RSX-11M send/receive directives. For further information on mailboxes, see the VAX-11/RSX-11M Programmer's Reference Manual for RSX-11M images or the VAX/VMS System Services Reference Manual for native images.
4. For RSX-11M task images issuing send/receive directives, the mailboxes used in emulating these directives are not created until just prior to actual image execution. Therefore, using either the /INTERVAL or /DELAY keywords does not cause creation of the mailboxes when the command is issued; you cannot assume that the mailboxes exist until image execution begins.
5. For an RSX-11M task image, the /INTERVAL keyword has the same effect as /DELAY; that is, the process execution occurs at the designated time, and when the image exits, the process is deleted. The image is executed only once.

## MCR COMMANDS

6. For a native image, /INTERVAL causes the image to hibernate and be re-executed after the specified interval if the image issues a VAX-11 MACRO instruction, RET (Return from Procedure). The image exits, however, if it issues an Exit system service.
7. Use the following keywords to assign equivalence names for the logical names SYS\$INPUT (TI), SYS\$OUTPUT (TI), and SYS\$error (CL):

```
/INPUT
/OUTPUT
/ERROR
```

8. Use the following keywords to override the default attributes for a process:

```
/ACCOUNTING
/PRIORITY
/PRIVILEGES
/PROCESS_NAME
/SERVICE_FAILURE
/SWAPPING
```

9. When you issue a RUN command and a detached process is created, pooled quotas are established which restrict the amount of various system resources available to the process and its descendent process(es). The following resource quotas are pooled when you create a detached process; that is, the values you specify are deducted from your current quotas and given to the detached process:

```
/BUFFER_LIMIT
/FILE_LIMIT
/PAGE_FILE
/QUEUE_LIMIT
/SUBPROCESS_LIMIT
```

The quota amounts are returned to you when the detached process is deleted.

10. The system defines minimum values for each specifiable quota; if you specify a quota that is below the minimum, or if you specify a deductible quota that reduces your quota below the minimum, the RUN command cannot create the process. To determine your current quotas, issue the SHOW PROCESS/QUOTAS command.
11. You also can specify limits that affect performance but are not deducted from your process's resources. The nondeductible quotas are:

```
/AST_LIMIT
/MAXIMUM_WORKING_SET
/IO_BUFFERED
/IO_DIRECT
/WORKING_SET
```

12. Hibernation is the VAX/VMS concept that corresponds to the RSX-11M concept of a suspended task.

## MCR COMMANDS

13. Use the following keywords to schedule execution of the image:

```
/DELAY
/INTERVAL
/SCHEDULE
```

If you specify any of these keywords, the RUN command creates the process, places it in a state of hibernation, and schedules a wake-up request for the appropriate time. The process cannot execute the image until it is wakened. For RSX-11M task images, /INTERVAL has the same effect as /DELAY.

14. Issue a STOP command to terminate execution of the image in the process and, if the process is not your current process, this process is deleted.
15. Issue a CANCEL command to cancel wake-up requests that are scheduled for the process but not yet delivered.
16. When a command accepts a keyword that specifies a time value, the time value is an absolute time, a delta time, or a combination of the two. The syntax rules for specifying time values are described in Section 2.7.6 and summarized below.

- Absolute times have the format:

```
[dd-mmm-yyyy[:]][hh:mm:ss.cc]
```

- Delta times have the format:

```
[dd-][hh:mm:ss.cc]
```

- Combination absolute and delta times have the format of an absolute time plus (+) or minus (-) a delta time.

## SET

## 4.28 SET

The SET command provides a number of options that allow you to set characteristics and defaults for your process, job, and certain devices. Table 4-1 briefly summarizes all the SET options. Note that RSX-11M MCR supports only the /UIC set command option.

Most of the SET options, which are summarized below, are equivalent to those described in detail in the VAX/VMS Command Language User's Guide or the VAX/VMS System Management and Operations Guide. Each summary below references the manual in which a full description of the option can be found.

Table 4-1: SET Command Options

| Option                     | Function                                                                                                                       |
|----------------------------|--------------------------------------------------------------------------------------------------------------------------------|
| ACCOUNTING <sup>2</sup>    | Selectively enables or disables the recording of particular kinds of accounting information                                    |
| CARD_READER <sup>2</sup>   | Defines the translation mode for a card reader                                                                                 |
| [NO]CONTROL_y <sup>2</sup> | Controls whether the use of the CTRL/Y function key can interrupt an image                                                     |
| DAY <sup>2</sup>           | Changes the default day type, which controls when certain users can or cannot log in                                           |
| DEFAULT <sup>2</sup>       | Establishes a disk and/or directory as the current default                                                                     |
| DEVICE <sup>2</sup>        | Establishes the spooling and error-logging status on a device                                                                  |
| DIRECTORY <sup>2</sup>     | Modifies the characteristics of a directory                                                                                    |
| FILE <sup>2</sup>          | Modifies the characteristics of a file                                                                                         |
| HOST <sup>1</sup>          | Establishes a virtual communication link between a terminal and a network node to which the terminal is not directly connected |
| LOGINS <sup>2</sup>        | Establishes the maximum number of users able to log in to the system                                                           |

1. Described in this manual.

2. Described in the VAX/VMS Command Language User's Guide.

(continued on next page)

## MCR COMMANDS

Table 4-1 (Cont.): SET Command Options

| Option                          | Function                                                                                                                                  |
|---------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------|
| MAGTAPE <sup>2</sup>            | Defines the density of a magnetic tape device or rewinds the tape                                                                         |
| MESSAGE <sup>2</sup>            | Overrides or supplements system messages                                                                                                  |
| [NO]ON <sup>2</sup>             | Controls whether the command interpreter checks for an error condition following the execution of a command in a command procedure        |
| PASSWORD <sup>1</sup>           | Allows users to change their own passwords                                                                                                |
| PRINTER <sup>2</sup>            | Defines characteristics of a printer                                                                                                      |
| PROCESS <sup>2</sup>            | Changes execution characteristics of a process                                                                                            |
| PROCESS/PRIORITY <sup>2</sup>   | Changes the base priority of a process                                                                                                    |
| PROTECTION <sup>2</sup>         | Changes the protection applied to a file or a group of files, restricting or allowing access to the file by different categories of users |
| PROTECTION/DEFAULT <sup>2</sup> | Establishes the default protection for all files subsequently created during the terminal session or batch job                            |
| PROTECTION/DEVICE <sup>2</sup>  | Established the protection for a non-file-structured device                                                                               |
| QUEUE/ENTRY <sup>2</sup>        | Changes the attributes associated with one or more entries in a print or batch job queue                                                  |
| RMS_DEFAULT <sup>2</sup>        | Provides default multiblock and multibuffer count values to be used by VAX-11 RMS for file operations                                     |
| TERMINAL <sup>2</sup>           | Changes characteristics of a terminal                                                                                                     |
| TIME <sup>2</sup>               | Resets the system time                                                                                                                    |
| UIC <sup>1</sup>                | Changes the UIC of the current process                                                                                                    |
| [NO]VERIFY <sup>2</sup>         | Controls whether the command interpreter displays lines in command procedures as it executes them                                         |
| VOLUME <sup>2</sup>             | Changes characteristics of one or more Files-11 volumes                                                                                   |
| WORKING_SET <sup>1</sup>        | Establishes a default working set size for images executed in the current process                                                         |

1. Described in this manual.

2. Described in the VAX/VMS Command Language User's Guide.

## SET HOST

## 4.28.1 SET HOST

The SET HOST command connects your terminal (through the current host processor) to another processor, called the remote processor. The remote processor must be a VAX-11. This command can only be used with DECnet-VAX, not DECnet-11.

Use the SET HOST command to connect to another VAX-11 processor on a network. (Use the SHOW NETWORK command described in the VAX/VMS Command Language User's Guide to obtain the names of nodes readable to your node.) Once the connection is made, the remote processor prompts for the user name and password. You must have an account on the remote processor to log in.

Once you have connected to the remote processor and logged in, you can use VAX/VMS MCR or DCL commands just as you would on your local processor. You can even use the SET HOST command to connect to another remote processor, and so on.

Use the LOGOUT command to log off the last processor you have logged in on. If you have connected to and logged in on more than one processor, the LOGOUT command leaves you logged in on the next-to-last processor.

For example, if your local node is OHIO you can use SET HOST BLUE to connect to the node BLUE; you can then use SET HOST GREEN to connect (still through OHIO and BLUE) to the node GREEN.

If you then use the LOGOUT command, you have logged off (and disconnected from) the processor at node GREEN, but you are still logged in on (and connected to) the processor at BLUE. A second LOGOUT command logs you off and disconnects you from BLUE. A third LOGOUT command logs you off your local processor node, OHIO.

**Format:**

```
SET HOST node-name
```

node-name                    Specifies the node name for the remote processor.

**Example:**

```
> SET HOST PENN
Username: ADAMS
Password:
```

This SET HOST command connects the user terminal to the processor at the network node named PENN. The remote processor then prompts for user name and password. Use the normal login procedure to log in on the remote processor.

**Note:**

1. RSX-11M MCR does not support the SET HOST command option.

## SET PASSWORD

### 4.28.2 SET PASSWORD

The SET PASSWORD command allows users to change their own passwords.

To maintain secrecy, users may need to change their passwords from time to time. The SET PASSWORD command offers a means of doing this. However, the system manager can control which users have the right to change their passwords.

Passwords may contain from 1 to 31 characters. The valid characters are:

```
A-Z
a-z
0-9
$ (dollar sign)
_ (underscore)
```

When the old and new passwords are entered, the user input is not echoed (to help ensure secrecy). To protect against typing errors that might not be seen when entering the new password, you must enter the desired new password twice.

If an error occurs, the password remains unchanged.

Passwords that exceed six characters are recommended for improved security.

#### Format:

```
SET PASSWORD
```

#### Example:

```
> SET PASSWORD
Old password:
New password:
Verification:
```

In response to the SET PASSWORD command, the system requests the old password, then the new password. Then the system asks for the new password again for verification purposes. If the user is authorized to change this account's password, the old password is given correctly, and the new password is given identically twice, the password is changed. Otherwise, an error message appears and the password remains unchanged.

#### Note:

1. RSX-11M MCR does not support the SET PASSWORD command option.

**SET /UIC****4.28.3 SET /UIC**

The SET /UIC command establishes a new directory default and, if you have Change Mode to Kernel (CMKRNL) privilege, establishes a new user identification code (UIC) for your process.

**Format:**

SET /UIC=[g,m]

- g** Specifies the UIC group number in the range of 0 through 377 (octal).
- m** Specifies the UIC member number in the range of 0 through 377 (octal).

**Example:**

> SET /UIC=[200,200]

This command establishes [200,200] as the default directory and UIC for the issuing process.

**Notes:**

1. RSX-11M MCR supports the SET /UIC command.
2. To display your UIC, use the SHOW PROCESS command described in the VAX/VMS Command Language User's Guide.
3. If you do not have Change Mode to Kernel (CMKRNL) privilege, a warning message is issued to inform you that your user identification code default is unchanged. However, your directory default is always changed.

## SHOW

### 4.29 SHOW

The SHOW command provides a number of options that let you display characteristics and defaults for your process, jobs, and devices. Table 4-2 briefly summarizes all the SHOW options. Note that RSX-11M does not support any of the SHOW command options.

Since the SHOW command options are equivalent to the options of the DCL command SHOW, these options are described in the VAX/VMS Command Language User's Guide and only are summarized in Table 4-3 below.

Table 4-2: SHOW Command Options

| Option     | Displays                                                                                                                     |
|------------|------------------------------------------------------------------------------------------------------------------------------|
| [DAY]TIME  | The current date and time                                                                                                    |
| DEFAULT    | The current default disk and directory device                                                                                |
| DEVICES    | The status of devices in the system                                                                                          |
| ERROR      | The error count for the CPU, memory, and any devices with an error count greater than 0                                      |
| LOGICAL    | Current logical name assignments                                                                                             |
| MAGTAPE    | Information about a designated magnetic tape unit                                                                            |
| MEMORY     | Information about the use of physical memory and other system-wide resources                                                 |
| NETWORK    | The availability of network nodes, including the current node                                                                |
| PRINTER    | Information about a designated magnetic tape unit                                                                            |
| PROCESS    | Attributes of the current process, including privileges, resource quotas, memory usage, priority, and accounting information |
| PROTECTION | The current default protection applied to files                                                                              |
| QUEUE      | Print or batch jobs that have been queued but not yet printed or processed                                                   |
| QUOTA      | Displays the current disk quota that is authorized and used by a specific user on a specific disk                            |

(continued on next page)

Table 4-2 (Cont.): SHOW Command Options

| Option      | Displays                                                                                                    |
|-------------|-------------------------------------------------------------------------------------------------------------|
| RMS_DEFAULT | The current default multiblock and multibuffer counts used by VAX-11 RMS for file operations                |
| STATUS      | The status of the current job, including accumulated CPU time, open file count, and count of I/O operations |
| SYMBOL      | Current symbol definitions                                                                                  |
| SYSTEM      | A list of all processes in the system and related process information                                       |
| TERMINAL    | The device characteristics of your terminal                                                                 |
| TRANSLATION | The result of translating a logical name                                                                    |
| USERS       | A list of the current users, in ascending order of their terminal device names                              |
| WORKING_SET | The current working set size limit and quota                                                                |

## STOP

### 4.30 STOP

The STOP command terminates execution of one of the following:

- A program or indirect command file that was interrupted by CTRL/C or CTRL/Y.
- A program that is executing in a subprocess or detached process.

The STOP command causes an abnormal termination of the image currently executing; if the image has declared any exit handling routines, they are not given control.

Note that when an image is interrupted by CTRL/Y, and the RUN command is issued to execute another image, the interrupted image is also terminated. However, in this case exit handling routines are allowed to execute before the next image is run.

If you interrupt a command procedure by CTRL/Y and you issue the STOP command, or if the STOP command is executed in a command procedure, all command levels are unstacked and control returns to command level 0.

If you specify a process name or process identification, the STOP command terminates the image currently executing in the specified process and deletes the process. If the process is a batch job process, no notification of deletion occurs; the log file for the batch job does not print.

The GROUP user privilege is required to stop other processes in the same group. The WORLD user privilege is required to stop any process in the system.

For more information on how to use commands like this one in command procedures, consult the VAX/VMS Guide to Using Command Procedures.

#### Format:

```
STOP [keyword] [process-name]
```

|              |                                                                                                                                                                                                              |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| keyword      | /IDENTIFICATION=process-id                                                                                                                                                                                   |
| process-name | Specifies 1- through 15-alphanumeric character string name of the process to be deleted. The specified process must have the same group number in its user identification code (UIC) as the current process. |

If you specify the /IDENTIFICATION keyword the process name is ignored. If you specify neither the process-name parameter nor the /IDENTIFICATION keyword, the image executing in the current process is terminated.

**Command Keyword:**

/IDENTIFICATION=process-id

Specifies the process identification the system assigned to the process when the process was created. When you create a process with the RUN command, the RUN command displays the process identification number of the process it creates.

When you specify the process identification, you can omit leading zeros.

**Examples:**

```
^> RUN MYPROG
^Y
> STOP
```

The RUN command begins executing the image MYPROG. Subsequently, CTRL/Y interrupts the execution and the STOP command terminates the image.

```
> @TESTALL
^Y
> STOP
```

The @ (Execute Procedure) command executes the procedure TESTALL.COM. Pressing CTRL/Y interrupts the procedure and the STOP command returns control to the VAX/VMS MCR command interpreter.

```
^> RUN/PROCESS_NAME=LIBRA LIBRA
%RUN-S-PROC_ID, identification of created process is 0013340D
.
.
.
> STOP LIBRA
```

The RUN command creates a subprocess named LIBRA to execute the image LIBRA.EXE. Subsequently, the STOP command forces the image to exit and deletes the process.

```
> ON ERROR THEN STOP
.
.
.
```

In a command procedure, the ON command establishes a default action when any error occurs as a result of a command or program execution. The STOP command stops all command levels; if the ON command is executed in a command procedure that is executed from within another procedure, control does not return to the outer procedure, but to the command interpreter.

```
> @WARHOL.CMD
.
.
.
^Y
> STOP
```

This sequence starts the execution of the indirect file WARHOL.CMD, interrupts its execution by pressing CTRL/Y, and terminates its execution by typing STOP. By default, the image in the current process is terminated.

## MCR COMMANDS

```
> RUN /PROCESS=PRIOR LIMNER.TSK
:
:
^y
> STOP PRIOR
```

This sequence starts the execution of the image LIMNER.TSK in a subprocess named PRIOR. Typing STOP causes the image to be terminated and the subprocess to be deleted.

### Notes:

1. The current process does not require privilege to stop one of its subprocesses. The current process must have GROUP or WORLD privilege to stop any other process.
2. The STOP command causes an abnormal termination of the image currently executing; if the image has declared any exit handling routines, they are not given control.  
  
When an image is interrupted by CTRL/Y or CTRL/C, and another image is run, the interrupted image is also terminated. However, in this case, exit handling routines are allowed to execute before the next image runs.
3. When an indirect command file is stopped, the entire procedure (that is, all levels) is terminated and control returns to the command interpreter.
4. Stopping the image executing in a subprocess or detached process causes the specified subprocess or detached process to be deleted.
5. When you create a subprocess or detached process using the RUN command, the command displays the process identification of the process created.
6. The SHOW PROCESS command displays the current status of subprocesses.
7. RSX-11M MCR does not support the same functionality provided by the VAX/VMS MCR STOP command.

## STOP/ABORT

### 4.31 STOP/ABORT

The STOP/ABORT command aborts a job that is currently being printed. The /ABORT keyword is required.

Use this command only to abort the printing of jobs entered in the system output queues, that is, line printer or terminal queues.

When you issue the STOP/ABORT command, the job currently being printed is terminated, and the next job in the queue is dequeued, provided you have sufficient privileges to do so. You can always abort your own job, and you can even abort jobs of other users in your group if you have the GROUP user privilege. Otherwise, you need WORLD or OPERATOR user privilege to abort a job that is not your own.

For more information on how to use commands like this one in command procedures, consult the VAX/VMS Guide to Using Command Procedures.

**Format:**

```
STOP/ABORT printer-name[:]
```

printer-name[:] Specifies the name of the printer queue in which the job was entered.

**Example:**

```
> STOP/ABORT LPA1:
```

This command aborts the job currently printing on line printer LPA1.

**Note:**

1. RSX-11M MCR does not support the STOP/ABORT command.

## STOP/ENTRY

### 4.32 STOP/ENTRY

The STOP/ENTRY command deletes an entry from a batch queue while it is running. The /ENTRY keyword is required.

Use this command to terminate the execution of a batch job while it is running. This command cannot, however, delete a job while it is waiting to be executed. If you want to delete an entry from a device or batch job queue while the entry is waiting to be executed, use the DELETE/ENTRY command.

Note that you can always stop your own job while it is running. You can even stop execution of jobs of other users in your group if you have the GROUP user privilege. Otherwise, you need WORLD or OPERATOR user privilege to stop a running job that is not your own.

**Format:**

STOP/ENTRY=job-number queue-name[:]

|               |                                                                         |
|---------------|-------------------------------------------------------------------------|
| job-number    | Specifies the job number of the job to be deleted from the batch queue. |
| queue-name[:] | Specifies the name of the queue in which the job was entered.           |

**Example:**

> STOP/ENTRY=230 SYS\$BATCH

The STOP/ENTRY command deletes the job associated with the entry number 230 in the batch queue SYS\$BATCH.

**Note:**

1. RSX-11M MCR does not support the STOP/ENTRY command.

## STOP/REQUEUE

### 4.33 STOP/REQUEUE

Stops the printing of the job currently being printed and places that job at the end of the output queue.

When you requeue a job, that job is placed at the end of the queue with its priority level lowered to 1. The next job in the queue is immediately dequeued for printing.

This command is useful when the line printer runs out of paper while it is printing a job; or when a large job of low priority is currently printing one or more other jobs in the queue must be printed immediately.

Note that you can always requeue your own job. You can even requeue jobs belonging to other users in your own group if you have the GROUP user privilege. Otherwise, you need WORLD or OPERATOR user privilege to requeue a job that is not your own.

For more information on how to use commands like this one in command procedures, consult the VAX/VMS Guide to Using Command Procedures.

#### Format:

```
STOP/REQUEUE queue-name[:]
```

queue-name[:] Specifies the name of the queue to be stopped.

#### Example:

```
> STOP/REQUEUE LPB0:
```

This command suspends the current print operation on LPB0: and places the job that was currently printing at the end of the queue with a priority of 1. The print operation resumes by starting the next job in the queue.

#### Note:

1. RSX-11M MCR does not support the STOP/REQUEUE command.

## SUBMIT

### 4.34 SUBMIT

The SUBMIT command queues an MCR indirect command file or a DCL command procedure for execution as a batch job. Your default (not current) command interpreter determines whether you can submit an MCR indirect command file or a DCL command procedure as a batch job. If your default command interpreter is MCR, you can submit MCR indirect command files for batch processing; if it is DCL, you can submit DCL command procedures for batch processing. All output from the indirect command file is queued to the device assigned the logical name SYS\$PRINT at the time the job terminates unless you specify otherwise.

You should specify the file type in file specifications used for the SUBMIT command, especially if your default command language is MCR. Regardless of your default command language, if the submitted command file activates any nested command files, the directory name may need to be present as part of the file specification when the nested command files are specified (see Notes that follow).

A file(s) queued by the SUBMIT command are considered a job. The system assigns a unique job number to each job in the system. When you submit a batch job, the system displays both the job number it assigned to the job and the name of the batch job queue in which it entered your job.

**Batch Job Output:** When you submit command procedures for processing by the SUBMIT command, all output from the command procedure is written to a file called name.LOG where name is the file name of the first command procedure file in the job. (Use the /NAME keyword to give the job a different name.) This file is initially written on your default disk; when the batch job completes, the system queues the file to SYS\$PRINT and deletes the file after it has printed unless the /NOPRINT keyword is used, in which case the log file is neither printed nor deleted.

If multiple procedures are submitted, the job terminates if any procedure exits with an error or fatal error status.

For a description of creating and submitting batch jobs, see the VAX/VMS Guide to Using Command Procedures.

#### Format:

```
SUBMIT [keywords] file-spec [keyword][,...]
```

```
keywords /AFTER=time
 /CPUTIME=n
 /[NO]DELETE
 /[NO]HOLD
 /[NO]IDENTIFY
 /[NO]KEEP
 /[NO]LOG_FILE=[file-spec]
 /NAME=job-name
 /[NO]NOTIFY
 /PARAMETERS=parameters[,...]
 /[NO]PRINTER[=queue-name]
 /PRIORITY=n
 /QUEUE=queue-name[:]
```

## MCR COMMANDS

/REMOTE  
/WSDEFAULT=n  
/WSEXTENT  
/WSQUOTA=n

### file-spec

Specifies the name of a command procedure to be submitted for batch execution. If no file type is specified, the SUBMIT command uses the default file type of COM. If you specify more than one file, use either a comma (,) or a plus (+) to separate file specifications. Files are processed in the order of specification.

If the file specification contains a network node name, the /REMOTE keyword must be specified.

Full wild card characters are allowed in the file specification. See the section on file specifications in the VAX/VMS Command Language User's Guide for detailed information on wild card characters.

### Command Keywords:

#### /AFTER=time

Requests that the job be processed after a specific time of day. Specify the time according to the rules for specifying absolute times or a combination of absolute and delta times in Section 2.7.6.

If the specified time has already passed, the job is queued for immediate processing.

By default, jobs are submitted immediately.

#### /CPUTIME=n

Defines a CPU time limit for the batch job. You may specify a delta time (Section 2.7.6), the value 0, or the words NONE or INFINITE for n.

You cannot request more CPU time than permitted by the base limits or the value in your user authorization file. However, you can use this keyword to override the base queue value established by the system manager or the value in your user authorization file, when you need less CPU time than authorized.

If the CPU time is not restricted by the base queue limits or the value in your user authorization file, you can specify 0 or INFINITE to request an infinite amount of CPU time. Specify NONE when you want the amount of CPU time to default to your user authorization file value or the limit specified on the queue.

#### /HOLD /NOHOLD (default)

Controls whether the job is to be made available for immediate processing. If you specify /HOLD, the job is not released until you issue a SET QUEUE/NOHOLD command for it.

## MCR COMMANDS

`/IDENTIFY (default)`  
`/NOIDENTIFY`

Controls whether the command interpreter displays the job identification assigned to the job and the name of the device to which it has been queued, as follows:

JOB jobid ENTERED ON QUEUE device-name

Usually, the device-name is SYSS\$BATCH.

`/KEEP`  
`/NOKEEP (default)`

Controls whether the log file is deleted after it is printed.

`/LOG_FILE=file-spec`  
`/NOLOG_FILE`

Controls whether a specified log file is created for the job. By default, the log file is written to a file in your default directory with the same name as your command file and a file type of LOG. If you specify `/NOLOG_FILE`, the log file is written to the null device.

If the file specification contains a device name, the process executing the batch job must have access to that device.

`/NAME=job-name`

Defines a 1- to 8-character alphanumeric name string to identify the job. The name string is displayed in response to the `SHOW QUEUE` command described in the VAX/VMS Command Language User's Guide.

If you do not specify a name, the name string defaults to the file name of the first file, truncated to eight characters, if necessary.

Output files from the indirect command file are named job-name.LOG.

`/NOTIFY`  
`/NONOTIFY (default)`

Controls whether a message is broadcast to any terminal at which you are logged in, notifying you that your job has been completed or aborted.

`/PARAMETERS=parameters`

Specifies from one to eight optional parameters to be passed to the indirect file. The parameters define values that are equated to the symbols P1, P2, ... P8 in the indirect command file. The symbols are local to the specified indirect command file.

If you specify more than one parameter, separate them with commas and enclose them in parentheses.

The commas delimit the parameters. To specify a parameter that contains special characters or delimiters, enclose the parameter in quotation marks.

## MCR COMMANDS

**/PRINTER[=queue-name]  
/NOPRINTER**

Controls whether the job log is queued for printing when your job is completed. The **/PRINTER** keyword allows you to specify a particular print queue. By default, if you omit this keyword, the job log is printed on **SYSS\$PRINT**.

If you specify **/NOPRINT**, the **/KEEP** keyword is assumed.

**/PRIORITY=n**

Specifies the priority for the job being submitted. The value of **n** must be in the range of 0 through 31; 0 is the lowest priority, and 31 is the highest.

By default, jobs are assigned the same priority as your process. You can assign lower priorities to nonessential jobs to allow your other jobs to be processed sooner.

The **ALTPRI** privilege is required to set a priority value that is higher than your process's priority.

**/QUEUE=queue-name[:]**

Specifies the name of a specific batch job queue to which the job is to be submitted.

**/REMOTE**

Indicates that the specified command procedure be executed on a remote node. The file specification must contain the name of the node on which the file resides and at which the procedure is to be executed. See the DECnet-VAX User's Guide.

If you specify **/REMOTE**, you cannot specify any other keywords.

**/WSDEFAULT=n**

Defines a working set default for the batch job. You may specify a positive integer in the range 1 through 65535, 0, or the word **NONE** for **n**.

Use this keyword to override the base queue value established by the system manager or the value in your user authorization file, provided you want to impose a lower value. You may not request a higher value than is specified in your user authorization file.

Specify 0 or **NONE** if you want the working set value to default to either your user authorization file or the working set default specified on the queue.

**/WSEXTENT=n**

Defines a working set default for the batch job. You may specify a positive integer in the range 1 through 65, 535, 0, or the word **NONE** in place of **n**.

Use this keyword to override the base queue value established by the system manager or the value authorized in your user authorization file, provided you want

## MCR COMMANDS

to impose a lower value. Specify 0 or NONE if you want the working set extent defaulted to either your user authorization file or the working set extent specified on the queue. However, you may not want to request a higher value than your default.

**/WSQUOTA=n**

Defines the maximum size working set for the batch job. This is the working set quota. You may specify a positive integer in the range 1 through 65535, 0, or the word NONE for n.

Use this keyword to override the base queue value established by the system manager or the value in your user authorization file, provided you want to impose a lower value. You may not request a higher value than your default.

Specify 0 or NONE if you want the working set quota to default to either your user authorization file or the working set quota specified on the queue.

### File Keyword:

**/DELETE**  
**/NODELETE**  
(default)

Controls whether files are deleted after processing. If you specify the /DELETE keyword after the SUBMIT command name, all files in the job are deleted. If you specify the /DELETE keyword following a file specification, only the associated file is deleted after it is processed.

The protection code on the input file(s) must allow delete access to the default user identification code (UIC) of the user who submitted the job.

### Examples:

```
> SUBMIT AVERAGE
 Job 112 entered on queue SYSS$BATCH
```

The SUBMIT command enters the procedure AVERAGE.COM in the batch job queue. When the batch job completes, the log file AVERAGE.LOG is queued for printing.

```
> SUBMIT BACKUP/PARAMETERS=(TXT,DOC,MEM), -
 > AVERAGE, RUNMASTER
 Job 416 entered on queue SYSS$BATCH
```

The SUBMIT command enters three command procedures in a single job. The job is given three parameters: P1 is equated to the string TXT, P2 to the string DOC and P3 to the string MEM. After the procedure BACKUP.COM is executed, the procedures AVERAGE.COM and RUNMASTER.COM are executed.

```
> BATCH/NAME=BATCH_24/HOLD TESTALL
 Job 467 entered on queue SYSS$BATCH
```

The SUBMIT command enters the procedure TESTALL.COM for processing as a batch job, but in a HOLD status. The job will not be released until

## MCR COMMANDS

the SET QUEUE/RELEASE command is issued. The /NAME keyword requests that the batch job be identified as BATCH\_24.

```
>SUBMIT SHEELER
JOB 134 ENTERED ON QUEUE "SYS$BATCH"
```

This command submits the indirect command file SHEELER.COMD for processing. The job is assigned the job identification 134 and placed in the default batch queue, SYS\$BATCH.

```
>SUBMIT BURCHFIELD /PARAMETERS="DBA2:[PAINTER]NATURE.DAT"
JOB 123 ENTERED ON QUEUE "SYS$BATCH"
```

This command submits the indirect command file BURCHFIELD and defines the symbol P1 as being equal to the file specification DBA2:[PAINTER]NATURE.DAT.

### Notes:

1. When the indirect command file is dequeued for execution, it is processed by the default command interpreter of the user who submitted the job. VAX/VMS does not consider the command interpreter used to submit the job.
2. The file(s) queued for process using the SUBMIT command are considered a job. The system assigns a unique job identification to each job. You can use the SHOW QUEUE command described in the VAX/VMS Command Language User's Guide to determine the job identification of a queued batch job.
3. The attributes of a queued job can be changed using the SET QUEUE/ENTRY command as described in the VAX/VMS Command Language User's Guide.
4. You can use the DELETE/ENTRY command to remove a job from a batch queue or to delete an entry that is being processed.
5. RSX-11M MCR does not support the SUBMIT command (RSX-11M-PLUS MCR does).
6. Care is advised when you must rely on the default file types used by the SUBMIT command instead of including the file type as part of the file specification. For the file specification used for the SUBMIT command, the default file type is always .COM regardless of your current command language or default command language in the user authorization file.

If your default command language is DCL, you can only execute a DCL command procedure and the default file type is .COM. If your default command language is MCR, although you can only execute an MCR indirect command file, the default file type is .COM, not .CMD (use the RENAME command to change file names).

If the command file specified for the SUBMIT command activates a nested command file, the type of command file you can execute does not change. However, if your default command language is MCR, the default file type becomes .CMD, not .COM, and the directory searched is your default login directory, not your current directory. If your default command language is DCL, the default file type remains .COM, but the directory searched is your default login directory, not your current directory.

## TIME

### 4.35 TIME

The TIME command displays the current date and time. This command is equivalent to the SHOW DAYTIME and SHOW TIME commands described in the VAX/VMS Command Language User's Guide.

**Format:**

TIME

**Example:**

```
> TIME
22-JUN-1982 12:43:12
```

**Note:**

1. RSX-11M MCR supports the TIME command.

**TYPE****4.36 TYPE**

The TYPE command displays the contents of a file or group of files on the terminal.

**Format:**

TYPE [keyword]file-spec [,...]

keyword                    /OUTPUT=file-spec

file-spec [,...]        Is the file specification of one or more files to be displayed.

If multiple files are specified, they can be separated with either a comma (,) or a plus sign (+). In either case, the files are displayed in the order named.

You can use a wild card instead of the directory, file name, file type, or version fields. The TYPE command displays all files that satisfy the file description in alphanumeric order for Files-11 Structure Level 2. Files are displayed in random order for Files-11 Structure Level 1.

If you omit the file type, the TYPE command uses LIS by default.

**Command Keyword:**

/OUTPUT=file-spec        Requests that the output from the TYPE command be written to the specified file, rather than to the SYS\$OUTPUT device.

**Example:**

> TYPE POLLOCK

The TYPE command locates POLLOCK.LIS in the default directory and types it on the terminal.

**Notes:**

1. To temporarily halt the output, press CTRL/S. To resume output after pressing CTRL/S, press CTRL/Q.
2. To suppress the display while allowing processing to continue, press CTRL/O. If CTRL/O is pressed again before the command terminates, output resumes at the current point in command processing.

## MCR COMMANDS

If you have entered more than one file specification or used wild cards, CTRL/O suppresses the display of the current file only. Output is displayed again when the TYPE command begins the next file.

3. To stop the display completely, press CTRL/Y and then use the STOP command.
4. RSX-11M MCR does not support the TYPE command.

## UFD

## 4.37 UFD

The UFD command creates a user file directory or subdirectory on a Files-11 Structure Level 1 or 2 volume and enters the name of the UFD in the volume's master file directory. The volume must have been initialized and must be mounted before you can define UFDs for it.

To create a first-level user file directory, you must have write access to the master file directory on the volume where you are creating the directory. On a system volume, normally only users with a system user identification code (UIC) or the SYSPRV or BYPASS user privilege are allowed write access to the master file directory to create a first-level directory. To create a subdirectory, you must be allowed write access to the lowest level directory that currently exists. The protection and UIC for the master file directory are established during volume initialization; they are the volume protection and volume owner UIC.

## Format:

UFD ddcu:[directory] [/keywords]

keywords        /ALLOC=number-entries  
                   /PRO=[system,owner,group,world]  
                   /UIC=[g,m]

ddcu            Specifies the device, controller, and unit number of the device containing the volume.

[directory]    Specifies the directory or subdirectory name. If you create a subdirectory on a structure level 1 volume, RSX-11M cannot read it.

## Command Keywords:

/ALLOC=number-entries        Specifies the number of directory entries for which space is to be allocated. The number provided is rounded up to the next multiple of 32 (decimal).

If you do not specify /ALLOC, the command allocates space for 32 directory entries.

/PRO=[system,owner  
           group,world]        Establishes access rights for the directory file. Specify protection codes for each user category.

Read (R), write (W), and delete (D) protection are identical when the volume is being accessed under either VAX/VMS or RSX-11M. The E field indicates execute under VAX/VMS and extend under RSX-11M.

## MCR COMMANDS

If you do not specify /PRO, the command supplies the default protection of the directory (MFD, UFD, or subdirectory) in which the new directory is being created.

If you omit any one of the protection categories, users in that category are denied access to the directory.

/UIC=[g,m]

Specifies the owner UIC for the directory or subdirectory. If you do not specify /UIC, the UFD command by default uses the owner UIC of the directory in which the new directory is being created.

### Example:

```
> MOU DBB2: MYVOL
> UFD DBB2:[230,222]/PRO=[RWED,RWED,RE,R]
```

The first command mounts the volume labeled MYVOL. The UFD command requests the creation of a directory with UIC [230,222] on the volume and establishes the protection for that directory.

### Notes:

1. VAX/VMS does not support the /UF switch for PIP to create directories.
2. RSX-11M MCR supports the UFD command.

## CHAPTER 5

### INDIRECT COMMAND FILES

VAX/VMS supports RSX-11M indirect command files to provide the same functions that RSX-11M Version 3.2 provides:

- Indirect command files for components, for example, MACRO-11 and the Task Builder
- MCR indirect command files

Indirect command files intended for processing by RSX-11M components function identically under VAX/VMS and RSX-11M, as described in Chapter 3. MCR indirect command files that are to execute under VAX/VMS can contain the following:

- Any of the indirect command file processor directives summarized alphabetically in Table 5-1
- Any of the VAX/VMS MCR commands described in Chapter 4

MCR indirect command files can be nested; refer to Section 5.14.

#### NOTE

The MCR command language does not support the level of functionality found in the RSX-11M Version 4.0 indirect command processor (ICP). For this reason, it is not possible to perform a NETgen for an RSX-11S or RSX-11M system under VAX/VMS. You can, however, perform an RSX-11S Version 4.0 system generation under VAX/VMS MCR.

Unlike RSX-11M, which provides an indirect command file processor task (AT.) to interpret directives, the VAX/VMS MCR command interpreter processes both MCR commands and directives. As a result, you can type directive sequences at the terminal and test them without creating and executing an indirect command file.

Your command interpreter must be MCR if you want to execute indirect command files. The login procedure described in Section 3.1.1 explains how to establish MCR as your command interpreter.

VAX/VMS support of indirect command files is intended to allow existing files to execute under VAX/VMS. Because it is intended for existing files, VAX/VMS does not perform such stringent syntax checking within the indirect command file as RSX-11M does. As a result, indirect command files developed under VAX/VMS may not execute successfully under RSX-11M.

Table 5-1: Summary of VAX/VMS Indirect Command File Processor Directives

| Format                                             | Function                                                                                    | Differences between VAX/VMS and RSX-11M         |
|----------------------------------------------------|---------------------------------------------------------------------------------------------|-------------------------------------------------|
| /                                                  | Closes all data and command files and exits                                                 | None                                            |
| .label:                                            | Defines a label                                                                             | None                                            |
| !text or ;text                                     | Provides commentary                                                                         | None                                            |
| .ASK ssssss txt-string                             | Asks a question and waits for a reply                                                       | None                                            |
| .ASKN ssssss txt-string                            | Asks for definition of a numeric symbol                                                     | None                                            |
| .ASKN[low:high] ssssss txt-string <sup>1</sup>     | Asks for definition of a numeric symbol specifying an allowable range                       | Can specify hexadecimal values; see Section 5.7 |
| .ASKN[:,def] ssssss txt-string <sup>1</sup>        | Asks for definition of a numeric symbol specifying the default value                        | Can specify hexadecimal values; see Section 5.7 |
| .ASKN[low:high:def] ssssss txt-string <sup>1</sup> | Asks for definition of a numeric symbol specifying an allowable range and the default value | Can specify hexadecimal values; see Section 5.7 |
| .ASKS ssssss txt-string                            | Asks for definition of a string symbol                                                      | None                                            |
| .ASKS[low:high] ssssss txt-string <sup>1</sup>     | Asks for definition of a string symbol specifying the allowable number of characters        | None                                            |
| .BEGIN                                             | Marks the beginning of a Begin-End block                                                    | None                                            |

1. Brackets are required syntax in these directives only. Brackets in other directives indicate optional features.

(continued on next page)

Table 5-1 (Cont.): Summary of VAX/VMS Indirect Command File Processor Directives

| Format                | Function                                                                | Differences between VAX/VMS and RSX-11M                                                         |
|-----------------------|-------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------|
| .CHAIN file-spec      | Continues processing using another file                                 | None                                                                                            |
| .CLOSE [#n]           | Closes secondary file                                                   | None                                                                                            |
| .DATA [#n] txt-string | Writes data to secondary file                                           | None                                                                                            |
| .DEC ssssss           | Decreases value of numeric symbol by 1                                  | None                                                                                            |
| .DELAY nnu            | Delays indirect command file execution for the specified period of time | VAX-11 hardware provides a 100-ticks-per-second clock; under VAX/VMS, hours cannot be specified |
| .DISABLE DATA [#n]    | Disables writing of data to secondary file                              | None                                                                                            |
| .DISABLE DOLLARS      | Disables dollar sign recognition                                        | RSX-11M does not support this directive; see Section 5.11                                       |
| .DISABLE ESCAPE       | Disables use of escape character as response to .ASK, .ASKN, or .ASKS   | None                                                                                            |
| .DISABLE GLOBAL       | Disables definition of global symbols                                   | None                                                                                            |
| .DISABLE LOWERCASE    | Enables lowercase-to-uppercase conversion                               | None                                                                                            |
| .DISABLE QUIET        | Suppresses echoing of CLI command lines                                 | Ignored by VAX/VMS; see Section 5.5                                                             |
| .DISABLE SUBSTITUTION | Disables symbol substitution                                            | None                                                                                            |

(continued on next page)

Table 5-1 (Cont.): Summary of VAX/VMS Indirect Command File Processor Directives

| Format               | Function                                                                                                                 | Differences between VAX/VMS and RSX-11M                   |
|----------------------|--------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------|
| .ENABLE DATA [#n]    | Enables writing of data to a secondary file                                                                              | None                                                      |
| .ENABLE DOLLARS      | Enables dollar sign recognition                                                                                          | RSX-11M does not support this directive; see Section 5.11 |
| .ENABLE ESCAPE       | Enables use of escape character as response to .ASK, .ASKN, or .ASKS                                                     | None                                                      |
| .ENABLE GLOBAL       | Enables definition of global symbols (Ssssss)                                                                            | None                                                      |
| .ENABLE LOWERCASE    | Enables use of lowercase characters in strings                                                                           | Embedded multiple tabs and spaces are not compressed      |
| .ENABLE QUIET        | Enables echoing of CLI command lines                                                                                     | Ignored by VAX/VMS; see Section 5.5                       |
| .ENABLE SUBSTITUTION | Enables symbol substitution                                                                                              | None                                                      |
| .END                 | Marks the end of a Begin-End block                                                                                       | None                                                      |
| .ERASE GLOBAL        | Erases global symbol definitions                                                                                         | Does not erase permanent global symbols                   |
| .ERASE LOCAL         | Erases local symbol definitions                                                                                          | None                                                      |
| .EXIT [n]            | Terminates processing of a Begin-End block or current command file and optionally sets the special symbol <EXSTAT> value | None                                                      |
| .GOSUB label         | Calls a subroutine                                                                                                       | None                                                      |
| .GOTO label          | Branches to a label                                                                                                      | None                                                      |

(continued on next page)

Table 5-1 (Cont.): Summary of VAX/VMS Indirect Command File Processor Directives

| Format                | Function                                             | Differences between VAX/VMS and RSX-11M                                       |
|-----------------------|------------------------------------------------------|-------------------------------------------------------------------------------|
| .IF sssss relop expr  | Determines whether symbols meet specified conditions | None                                                                          |
| .IF[N]ACT tttttt      | Determines whether a task is active                  | Response always indicates that the task is inactive; see Section 5.2          |
| .IF[N]DF ssssss       | Determines whether a symbol is defined               | None                                                                          |
| .IFF ssssss           | Determines whether a symbol is false                 | None                                                                          |
| .IF[N]INS tttttt      | Determines whether a task is installed               | Response always indicates that the task is already installed; see Section 5.2 |
| .IF[N]LOA dd:         | Determines whether a loadable driver is loaded       | Response always indicates that the driver is loaded; see Section 5.2          |
| .IFT ssssss           | Determines whether a symbol is true                  | None                                                                          |
| .INC ssssss           | Increases the value of a numeric symbol by 1         | None                                                                          |
| .ONERR label          | Branches to label on detecting an error              | Equivalent to an ON WARNING THEN .GOTO label command; see Section 5.3         |
| .OPEN [#n] file-spec  | Opens a secondary file for output                    | None                                                                          |
| .OPENA [#n] file-spec | Opens an existing file and appends subsequent data   | None                                                                          |

(continued on next page)

Table 5-1 (Cont.): Summary of VAX/VMS Indirect Command File Processor Directives

| Format                                     | Function                                                                                    | Differences between VAX/VMS and RSX-11M                                                                   |
|--------------------------------------------|---------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------|
| .OPENR [#n] file-spec                      | Opens a file for reading with the .READ directive                                           | None                                                                                                      |
| .PARSE string control-string var1 ... varn | Parses strings in a command line into substrings                                            | None                                                                                                      |
| .PAUSE                                     | Pauses for user action                                                                      | Use the CONTINUE command to resume indirect command file processing                                       |
| .READ [#n] string-variable                 | Reads the next record into the specified string variable                                    | Maximum record length is not 132 bytes as under RSX-11M; it is 255 bytes for this directive under VAX/VMS |
| .RETURN                                    | Returns from subroutine                                                                     | None                                                                                                      |
| .SETF ssssss                               | Sets the value of a symbol to false                                                         | None                                                                                                      |
| .SETN ssssss numexp                        | Sets a symbol to a numeric value                                                            | Can specify hexadecimal value; see Section 5.7                                                            |
| .SETS ssssss strexp                        | Sets a symbol to a string value                                                             | None                                                                                                      |
| .SETT ssssss                               | Sets the value of a symbol to true                                                          | None                                                                                                      |
| .STOP [n]                                  | Terminates indirect command file processing and optionally sets the special symbol <EXSTAT> | None                                                                                                      |
| .TEST ssssss                               | Tests the length of a string symbol                                                         | None                                                                                                      |
| .TESTFILE file-spec                        | Determines whether the specified file exists                                                | Sets <FILSPC> and <FILERR> but not <FILATR>                                                               |

### 5.1 REQUESTING INDIRECT COMMAND FILE EXECUTION

Under VAX/VMS, you request execution of an indirect command file in the same way that you do under RSX-11M; that is, by preceding the file specification of the indirect command file with an at sign (@). However, under VAX/VMS, additional options are available:

- You can use the /OUTPUT keyword to request that all output to SYS\$OUTPUT be written to a specific file or device.
- You can pass parameter values (P1 through P8) to the indirect file.

The Execute Procedure described in Chapter 4 contains a description of these options.

### 5.2 SUPPORT OF .IFINS, .IFACT, AND .IFLOA

The following directives test conditions that are not defined in the VAX/VMS environment:

- .IFINS and .IFNINS
- .IFACT and .IFNACT
- .IFLOA and .IFNLOA

VAX/VMS supports these directives by always making the following standard responses to them:

- Task is installed; that is, .IFINS is true and .IFNINS is false.
- Task is inactive; that is, .IFACT is false and .IFNACT is true.
- Driver is loaded; that is, .IFLOA is true and .IFNLOA is false.

Under VAX/VMS, users usually do not load drivers. Drivers normally are loaded when the system is initialized. If a particular driver is needed, but has not been loaded, ask the system manager for assistance.

### 5.3 SUPPORT OF .ONERR

The .ONERR label is equivalent to using the ON WARNING THEN .GOTO label command. See the ON Severity-level Statement described in Chapter 4 for more information.

### 5.4 UNSUPPORTED DIRECTIVES

VAX/VMS does not support the following indirect command file processor directives:

- .XQT and .WAIT
- .ONERR

## INDIRECT COMMAND FILES

- Any indirect command file processor directives added since RSX-11M Version 3.2.

All other directives are supported. The following sections describe differences in directive support.

### 5.4.1 .XQT and .WAIT Alternative

As an alternative to .XQT and .WAIT, you can use the RUN command. Because images execute serially in your process, the .WAIT directive is not needed; one image must terminate before the next begins.

If you use the RUN command to create a subprocess or detached process, there is no means of synchronizing image execution from the indirect command file.

### 5.4.2 .ONERR Alternative

The VAX/VMS ON statement, described in Chapter 4, can serve as an alternative to the .ONERR directive. However, the ON Statement provides a more general error handling capability than .ONERR. The MCR command interpreter does not restrict use of error handling to the specific errors associated with the .ONERR directive.

## 5.5 SWITCHES

RSX-11M defines three switches for use with indirect command file specifications:

- /TR (trace) and /NOTR
- /DE (delete) and /NODE
- /MC (pass command to MCR) and /NOMC

The defaults under RSX-11M are /NOTR, /NODE, and /MC. That is, indirect command files are not traced during execution and not deleted after processing, and the MCR command interpreter processes all commands.

VAX/VMS supports the default case and also allows use of the /DE switch to delete files after processing. It does not support the /TR and /MC switches. No deletion (/DE or /NODE) is the default.

## 5.6 SPECIAL SYMBOLS

VAX/VMS supports all the special symbols defined by RSX-11M. Symbol substitution for special symbols and user-defined symbols is the same under both RSX-11M and VAX/VMS MCR. Table 5-2 summarizes the symbols and provides VAX/VMS-specific information where appropriate.

INDIRECT COMMAND FILES

Table 5-2: Special Symbols

| Symbol    | Meaning                                                                                                   | VAX/VMS-Specific Information                        |
|-----------|-----------------------------------------------------------------------------------------------------------|-----------------------------------------------------|
| <ALPHAN>  | True if last string entered as answer to .ASKS or tested with .TEST contains only alphanumeric characters | --                                                  |
| <CLI>     | Returns the name of the current command language interpreter as an ASCII string                           | Always MCR                                          |
| <DATE>    | Assigned the current date; format is dd-mmm-yyyy                                                          | --                                                  |
| <DEFAULT> | True if answer to last .ASKN was defaulted                                                                | --                                                  |
| <ESCAPE>  | True if last query was answered with a single escape character                                            | --                                                  |
| <EXSTAT>  | Equivalent to \$SEVERITY                                                                                  | --                                                  |
| <LIBUIC>  | Nonprivileged task library UIC                                                                            | Always [1,54]                                       |
| <MAPPED>  | True if the system on which the image is running is mapped and false if unmapped                          | Always true                                         |
| <MEMSIZ>  | Memory size in K words                                                                                    | Always 32K                                          |
| <RAD50>   | True if last string entered as answer to .ASKS or tested with .TEST contains only Radix-50 characters     | --                                                  |
| <RSX11D>  | True if RSX-11D system                                                                                    | Always false                                        |
| <STRLEN>  | Length of last string entered as answer to .ASKS or tested by .TEST                                       | --                                                  |
| <SYDISK>  | Device mnemonic of system device                                                                          | Always SY                                           |
| <SYSTEM>  | Octal number representing operating system                                                                | Always 5 under VAX/VMS <sup>1</sup>                 |
| <SYSUIC>  | System UIC                                                                                                | Always [1,54]                                       |
| <SYUNIT>  | Unit number of system device                                                                              | Always 0                                            |
| <TIME>    | Assigned the current time; format is hh:mm:ss                                                             | --                                                  |
| <UIC>     | Current directory                                                                                         | The current directory is in a valid VAX/VMS format. |

1. Note that <SYSTEM> is 1 under RSX-11M and 6 under RSX-11M-PLUS. You can use <SYSTEM> to allow one indirect command file to run under multiple operating systems.

## 5.7 NUMERIC SYMBOLS

Under VAX/VMS, a numeric symbol defined using .SETN or .ASKN can be in one of the following radices:

- Octal (default radix)
- Decimal
- Hexadecimal

You can specify the radix of a numeric value in either of the following ways:

- Using RSX-11M conventions, that is, no radix indicator or a leading pound sign (#) for octal, and a trailing period for decimal
- Using a VAX/VMS radix indicator

VAX/VMS radix indicators have the format %rn. The percent sign (%) is required syntax; r indicates the radix (O for octal, D for decimal, and X for hexadecimal); and n is the value of the symbol. Regardless of the method used to specify the radix, the system stores and displays the value as a numeric string with the radix implied.

Examples:

```
> .SETN A 22
> .SETN B %X22
> .SETN C 22.
> .SETN D #007
> SHOW SYMBOLS/ALL
A = 22
B = 22
C = 22
D = 7
```

You cannot determine the radix of a symbol value using the SHOW SYMBOL command.

## 5.8 LABEL TABLE ENTRIES

As in RSX-11M, VAX/VMS places any label that appears on a line by itself in a label table so that, when a label is referred to, the command interpreter can locate it quickly. Labels defined on a line that also contains a command or other text are not placed in the label table but can be specified as the destination in .GOTO and .GOSUB directives.

The VAX/VMS MCR command interpreter establishes a new label table for each level of indirect command file. When a nested level exits, the current table is emptied to free space for the next higher level. Therefore, labels defined in an indirect command file must be unique within that file.

Labels passed over as a result of a .GOTO or .GOSUB directive are not placed in the label table. The label table contains only labels encountered in the actual execution stream.

## 5.9 SYSS\$INPUT AND SYSS\$COMMAND

VAX/VMS maintains two process-permanent files for command input: SYSS\$INPUT and SYSS\$COMMAND. SYSS\$INPUT is the prime source of command input. That is, for an interactive user, SYSS\$INPUT is the user's terminal; for indirect command file execution, SYSS\$INPUT is the file of commands and data. SYSS\$COMMAND is a secondary input source during execution of an indirect command file. It remains assigned to the initiating terminal. You can interrupt the processing of an indirect command file, issue a set of commands by means of SYSS\$COMMAND, and continue the indirect command file without altering its SYSS\$INPUT.

All of the MCR commands described in Chapter 4 have separate streams for SYSS\$INPUT and SYSS\$OUTPUT. However, to be compatible with RSX-11M, RSX-11M components such as PIP, EDI, TKB, and any other commands invoked by typing their image file name have SYSS\$INPUT assigned to SYSS\$COMMAND. That is, attempts to read from TI (SYSS\$INPUT) actually result in reading from SYSS\$COMMAND. This assignment allows the indirect command file to obtain information (for example, edits) from the terminal, as is often done under RSX-11M.

For cases in which the executing image is to read from SYSS\$INPUT (that is, when it expects to read records from the indirect command file), you can use the RUN command to request the image in the indirect command file. For example:

```
RUN SYSS$SYSTEM:PIP
```

## 5.10 .ASKN DIRECTIVE

When VAX/VMS executes an .ASKN directive, it displays the default radix of the value after the text string that requests the value. The default radix is displayed as [O] (octal), [D] (decimal), or [H] (hexadecimal). Similarly, the response can be octal, decimal, or hexadecimal. To override the default, you can follow the RSX-11M conventions (a leading pound sign forces octal and a trailing period forces decimal), or you can specify a VAX/VMS radix operator in the format %rn. No operator is required to specify a value in the default radix.

The response to the request for a symbol value can be a symbol name or an expression. If hexadecimal is the default radix and you want to enter a hexadecimal value that starts with A through F, you must enter a leading zero or use the radix operator %X to distinguish the hexadecimal value from a symbol name.

## 5.11 .ENABLE DOLLARS DIRECTIVE

Under RSX-11M, MCR indirect command files can contain only commands and directives. Under VAX/VMS, MCR indirect command files can contain commands, directives, and data. For example, an indirect command file can contain the command to run a program followed by the data read by that program.

The .ENABLE DOLLARS directive indicates to the VAX/VMS MCR command interpreter that the indirect command file is to take advantage of the command interpreter's capability to distinguish commands and directives from data. The command interpreter distinguishes between the two by checking for dollar signs (\$). Once dollar sign recognition is enabled, you must precede all subsequent command and directive lines with a dollar sign, for example, \$RUN.

## INDIRECT COMMAND FILES

The MCR command interpreter also uses the dollar sign convention to prevent user programs from reading past the end of data in the indirect command file and to bypass any data records that remain unread after the user program terminates. That is, if a user program does not read all its data, the command interpreter can discard the remaining data records and start processing the next command or directive.

The following is an example of a VAX/VMS MCR indirect command file containing data.

```
.ENABLE DOLLARS
$TIM
$RUN MYPROG
data to be read by MYPROG
.
.
.
$TIM
.DISABLE DOLLARS
```

By default, dollar sign recognition is disabled at the beginning of an indirect command file level.

### 5.12 .PAUSE DIRECTIVE

The .PAUSE directive functions the same way under VAX/VMS as it does under RSX-11M. That is, once the pause starts, you can run other images and utilities in your process. While in a pause condition, the process's SYS\$INPUT is equated to SYS\$COMMAND so that any images run during the pause can read input from the terminal. To continue from the pause, type the CONTINUE command.

If you interrupt image execution by pressing CTRL/Y or CTRL/C during a pause and then issue a CONTINUE command, the pause is terminated. When the image exits, the indirect command file resumes. If you issue a STOP command to terminate the image, the command interpreter closes the indirect command file and returns to interactive command level in addition to terminating the image.

### 5.13 .DELAY DIRECTIVE

Under VAX/VMS, you can use the .DELAY directive to delay indirect command file processing for a number of ticks, seconds, or minutes; you cannot specify a delay of a number of hours. To wait for an hour or more, specify an equivalent number of minutes, as follows.

```
.DELAY 90M
```

This directive causes a delay of 1 hour and 30 minutes.

Pressing CTRL/Y interrupts a delay. If you type a CONTINUE command, indirect command file processing resumes. A STOP command terminates the indirect command file.

## INDIRECT COMMAND FILES

### 5.14 NESTING MCR INDIRECT COMMAND FILES

The VAX/VMS MCR command interpreter allows the nesting of four indirect command files each having eight subroutines. As the number of subroutines decreases, the number of nested files allowed increases, and vice versa.

The number of nested indirect command files allowed by RSX-11M components running under VAX/VMS (for example, TKB) is controlled by individual components and is the same as under RSX-11M.

### 5.15 LEXICAL FUNCTIONS

VAX/VMS provides a set of lexical functions that return information about character strings and attributes of the current process. You can use lexical functions in MCR directives that accept expressions as parameters, for example, .SETN and .SETS. Table 5-3 summarizes the lexical functions supported for indirect command file use.

#### NOTE

Lexical functions added to VAX/VMS since Version 2.0 are not supported. See Table 5-3 for a list of supported lexical functions.

The VAX/VMS Command Language User's Guide provides a more detailed explanation of lexical functions; the following paragraphs summarize syntax rules.

The general format of a lexical function is:

```
'F$function=name([args,...])
```

'F Indicates that a function name follows. The substitution operator (' ) is required.

function-name Specifies the function to be returned. All function names are keywords. You can truncate function names to any unique truncation.

( ) Encloses function arguments, if any. The parentheses are required for all functions including those that do not accept arguments.

args,... Specify the arguments for the function. You can specify arguments using symbol names, numeric literals, or string literals enclosed in quotation marks.

Functions cannot be specified as arguments for functions. Literal strings specified as arguments cannot contain embedded symbols.

Section 5.17.2 contains an example of the use of lexical functions.

In the MCR command language, lexical functions are intended for use with the .SETx directives. If they are used in .IF directives, results may not be as expected. To make full use of lexical functions, you should use DCL rather than MCR.

INDIRECT COMMAND FILES

Table 5-3: Summary of Lexical Functions

| Function                            | Value Returned                                                                                                                                                            |
|-------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 'F\$DIRECTORY()                     | Current default directory name                                                                                                                                            |
| 'F\$EXTRACT(position,length,string) | Substring beginning in specified position for length specified of indicated string                                                                                        |
| 'F\$LENGTH(string)                  | Length of specified string                                                                                                                                                |
| 'F\$LOCATE(substring,string)        | Starting position of specified substring within string indicated; or, the length of the string if the substring is not found                                              |
| 'F\$LOGICAL(logical-name)           | Equivalence name of specified logical name (first match found in ordered search of process, group, and system logical name tables); or, logical name if no match is found |
| 'F\$MESSAGE(message-code)           | Message string associated with a system status value                                                                                                                      |
| 'F\$MODE()                          | Interactive or batch                                                                                                                                                      |
| 'F\$PROCESS()                       | Current process name                                                                                                                                                      |
| 'F\$TIME()                          | Current date and time of day                                                                                                                                              |
| 'F\$USER()                          | Current user identification code (UIC)                                                                                                                                    |
| 'F\$VERIFY()                        | TRUE (that is, a numeric value of 1) if verification is set on; FALSE (that is, a numeric value of 0) if verification is set off                                          |

5.16 OPERATIONAL NOTES

The notes below describe operational aspects of indirect command file processing under VAX/VMS.

- As in RSX-11M, all data files are closed when an indirect command file terminates and returns to the interactive command level, that is, when MCR prompts for the next command.
- When a command file terminates and returns to the interactive command level, the command interpreter clears the global symbol table.

## INDIRECT COMMAND FILES

- When lowercase is enabled, responses to .ASKS directives are treated as quoted strings; that is, the command interpreter does not change what is typed. As a result, embedded multiple tabs or spaces are not compressed.
- Under VAX/VMS, you cannot press CTRL/Z in response to a request for input (for example, .ASK) to terminate the indirect command file. You can, however, press CTRL/Y or CTRL/C to terminate the indirect command file at any point during processing.

### 5.17 RSX-11S SYSTEM GENERATION REQUIREMENTS

VAX/VMS uses the MCR indirect command file capability to serve as the host system for RSX-11S system generation. Requirements for RSX-11S system generation follow.

- You must have a command line interpreter symbol table of at least 40 pages. If the existing symbol table space is inadequate, you may reset the parameter CLISYMTBL in VAX/VMS system generation. (To invoke system generation, enter RUN SYSS\$SYSTEM:SYSGEN.) After you have changed this parameter, write it back to the current system image and reboot the system. (See the RSX-11S System Generation and Installation Guide).
- You must have the Change Mode to Kernel (CMKRNL) privilege, or the target disk volume must be unprotected. Change Mode to Kernel privilege is required to perform a SET /UIC command that changes the default UIC. Section 5.17.1 describes the creation of an unprotected disk volume.
- You must have logical I/O privilege (LOG\_IO) for VMR to perform logical I/O on system files.
- You must have system protection privilege (SYSPRV) which allows you to create files in other directories.
- You must have volume protection privilege (VOLPRO) if you are using a magnetic tape kit so you can mount the tape with the /FOREIGN keyword for the MOUNT command.

You can perform the RSX-11S system generation on the host system disk or on another disk. If you use the host system disk, you can select group numbers during the system generation procedure for the RSX-11S system and Executive files so that they will not conflict with files already on the disk. The valid group numbers you can select are between 3 and 10. The default group is 3.

If you use a disk other than the system disk, the valid group numbers are 1 through 10 for system files and 1 through 11 for Executive files.

Note that if you must choose alternative group numbers for the system and Executive files, you must explicitly specify the UFD in the command line whenever you use such files as the system object library (SYSLIB.OLB) or the system macro library (RSXMAC.SML). Normally, TKB and MACRO-11 expect to find these libraries in UFD [1,1].

Multiple RSX-11S system generations may be performed if there is sufficient room on the specified disk.

## INDIRECT COMMAND FILES

After you complete system generation, bring the system medium to an RSX-11S processor and boot it.

For more information on RSX-11S system generation, refer to the RSX-11S System Generation and Installation Guide.

## APPENDIX A

### PRIVILEGES

The following is a list of the privileges defined by VAX/VMS.

|              |                                                                                            |
|--------------|--------------------------------------------------------------------------------------------|
| [NO]ACNT     | Allows/disallows a process to disable accounting messages for created processes            |
| [NO]ALLSPOOL | Allows/disallows a process to allocate spooled devices                                     |
| [NO]ALTPRI   | Allows/disallows a process to set priority values                                          |
| [NO]BUGCHK   | Allows/disallows a process to make bug check error log entries                             |
| [NO]BYPASS   | Allows/disallows a process to bypass UIC protection in accessing files                     |
| [NO]CMEXEC   | Allows/disallows a process to change mode to executive                                     |
| [NO]CMKRNL   | Allows/disallows a process to change mode to kernel and change UICs                        |
| [NO]DETACH   | Allows/disallows a process to create detached processes                                    |
| [NO]DIAGNOSE | Allows/disallows a process to issue diagnostic I/O requests                                |
| [NO]EXQUOTA  | Allows/disallows a process to exceed resource quota                                        |
| [NO]GROUP    | Allows/disallows a process to control other processes in the same group                    |
| [NO]GRPNAM   | Allows/disallows a process to place names in the group logical name table                  |
| [NO]LOG_IO   | Allows/disallows a process to issue logical I/O requests to a device                       |
| [NO]MOUNT    | Allows/disallows a process to execute a Queue I/O Request system service to mount a device |
| [NO]NETMBX   | Allows/disallows a process to create a network device                                      |
| [NO]OPER     | Allows/disallows a process to have operator privileges                                     |
| [NO]PFNMAP   | Allows/disallows a process to map to specific physical pages                               |

## PRIVILEGES

|            |                                                                                |
|------------|--------------------------------------------------------------------------------|
| [NO]PHY_IO | Allows/disallows a process to issue physical I/O requests to a device          |
| [NO]PRMCEB | Allows/disallows a process to create permanent common event flag clusters      |
| [NO]PRMGBL | Allows/disallows a process to create permanent global sections                 |
| [NO]PRMMBX | Allows/disallows a process to create permanent mailboxes                       |
| [NO]PSWAPM | Allows/disallows a process to alter its swap mode                              |
| [NO]SETPRV | Allows/disallows a process to set other process's privileges                   |
| [NO]SHMEM  | Allows/disallows a process to create and/or delete structures in shared memory |
| [NO]SYSGBL | Allows/disallows a process to create system global sections                    |
| [NO]SYSLCK | Requests locks on system-wide resources.                                       |
| [NO]SYSNAM | Allows/disallows a process to place names in the system logical name table     |
| [NO]SYSPRV | Allows/disallows a process to attach system status for file access             |
| [NO]TMPMBX | Allows/disallows a process to create temporary mailboxes                       |
| [NO]VOLPRO | Allows/disallows a process to override volume protection                       |
| [NO]WORLD  | Allows/disallows a process to control all other processes in the system        |

APPENDIX B  
RESOURCE QUOTAS AND LIMITS

The following is a list of the resource quotas and limits defined by VAX/VMS.

|           |                                                                           |
|-----------|---------------------------------------------------------------------------|
| ASTLM     | Total outstanding AST operations plus scheduled wake-up requests          |
| BIOLM     | Total outstanding buffered I/O operations                                 |
| BYTLM     | Total outstanding buffer space quota                                      |
| CPUTIME   | Default CPU time limit in milliseconds (0 means infinite)                 |
| DIOLM     | Total outstanding direct I/O limit                                        |
| ENQLM     | Enqueue limit                                                             |
| FILLM     | Total outstanding open file quota                                         |
| PGFLQUOTA | Total pages in use in the system paging file                              |
| PRCLM     | Total outstanding subprocess                                              |
| TQELM     | Total outstanding entries in the timer queue plus temporary common events |
| WSDEFAULT | Default working set size                                                  |
| WSEXTENT  | Working set extent quota                                                  |
| WSQUOTA   | Maximum working set size                                                  |



## INDEX

- Abbreviation,
  - command name, 2-19
  - keyword, 2-19
- Absolute and delta time
  - combination,
    - example, 2-24
    - specification, 2-24
- Absolute time,
  - example, 2-23
  - keyword list, 2-22
  - specification, 2-21 to 2-23
- ALLOCATE command, 3-2 to 3-3, 4-2 to 4-3
- <ALPHAN> special symbol, 5-9
- ANSI (17-character) file name
  - support for magnetic tape, 1-8
- APPEND command, 4-4, 4-6
- ASG option for Task Builder, 3-12
- .ASK indirect command file directive, 5-2
- .ASKN indirect command file directive, 5-2, 5-10 to 5-11
- .ASKS indirect command file directive, 5-2, 5-15
- ASN command, 2-17, 3-4, 4-7 to 4-8
- ASSIGN LUN directive, 3-4 to 3-5
  
- Bad Block Locator Utility (BAD), 1-3
- .BEGIN indirect command file directive, 5-2
- Block,
  - mark the end of, 5-4
  - mark the start of, 5-2
- Branch to label indirect command file directive, 5-4 to 5-5
- BYE command, 4-9, 4-54
  
- Call subroutine indirect command file directive, 5-4
- CANCEL command, 4-10 to 4-11
- .CHAIN indirect command file directive, 5-3
- CL pseudodevice, 2-15 to 2-16
- <CLI> special symbol, 5-9
  
- Close file and exit indirect command file directive, 5-2
- .CLOSE indirect command file directive, 5-3
- CO pseudodevice, 2-15 to 2-16
- Combination absolute and delta time,
  - example, 2-24
  - specification, 2-24
- Command,
  - ALLOCATE, 3-2 to 3-3, 4-2 to 4-3
  - APPEND, 4-4, 4-6
  - ASN, 2-17, 3-4, 4-7 to 4-8
  - BYE, 4-9, 4-54
  - CANCEL, 4-10 to 4-11
  - CONTINUE, 4-12 to 4-13, 5-12
  - COPY, 4-14 to 4-17
  - DCL commands not found in MCR, 1-7
  - DEALLOCATE, 4-18
  - DEBUG, 3-7, 4-19
  - DELETE, 4-20 to 4-22
  - DELETE/ENTRY, 4-23 to 4-24
  - DEPOSIT, 4-25 to 4-27
  - DIRECTORY, 4-28 to 4-35
  - DMO, 4-36 to 4-37
  - equivalent function under VAX/VMS and RSX-11M, 1-4
  - EXAMINE, 4-25, 4-38 to 4-40
  - EXIT, 4-43 to 4-44, 4-66
  - file related, 1-6
  - informational, 1-5
  - initialization, 1-5
  - INITIALIZE, 4-45 to 4-51
  - invalid, 1-7
  - LOGOUT, 4-9, 4-54, 4-89
  - MOUNT, 3-3 to 3-4, 4-55 to 4-65
  - ON severity-level, 4-66 to 4-67
  - PRINT, 4-68 to 4-71
  - process control, 1-5
  - program development, 1-6
  - PURGE, 4-72 to 4-73
  - RENAME, 4-74 to 4-75
  - RUN (Image), 4-76
  - RUN (Process), 4-77 to 4-86
  - SET ACCOUNTING, 4-87
  - SET CARD READER, 4-87
  - SET DAY, 4-87
  - SET DEFAULT, 4-87
  - SET DEVICE, 4-87
  - SET DIRECTORY, 4-87
  - SET HOST, 4-89

## INDEX

- Command (Cont.)
  - SET LOGINS, 4-87
  - SET MAGTAPE, 4-88
  - SET MESSAGE, 4-88
  - SET [NO]CONTROL Y, 4-87
  - SET [NO]ON, 4-88
  - SET [NO]VERIFY, 4-88
  - SET ON, 4-66
  - SET PASSWORD, 4-90
  - SET PRINTER, 4-88
  - SET PROCESS, 4-88
  - SET PROCESS/PRIORITY, 4-88
  - SET PROTECTION, 2-21, 4-88
  - SET PROTECTION/DEFAULT, 4-88
  - SET PROTECTION/DEVICE, 4-88
  - SET QUEUE/ENTRY, 4-88
  - SET RMS DEFAULT, 4-88
  - SET TERMINAL, 4-88
  - SET TIME, 4-88
  - SET /UIC, 4-91
  - SET VOLUME, 4-88
  - SET WORKING SET, 4-88
  - SET[NO]ON, 4-67
  - SHOW [DAY]TIME, 4-92
  - SHOW DEFAULT, 4-92
  - SHOW DEVICES, 4-92
  - SHOW ERROR, 4-92
  - SHOW LOGICAL, 2-12, 4-92
  - SHOW MAGTAPE, 4-92
  - SHOW MEMORY, 4-92
  - SHOW NETWORK, 4-92
  - SHOW PRINTER, 4-92
  - SHOW PROCESS, 2-3, 4-92
  - SHOW PROTECTION, 4-92
  - SHOW QUEUE, 4-92
  - SHOW QUOTA, 4-92
  - SHOW RMS DEFAULT, 4-93
  - SHOW STATUS, 4-93
  - SHOW SYMBOL, 4-93
  - SHOW SYSTEM, 2-3, 4-93
  - SHOW TERMINAL, 4-93
  - SHOW TRANSLATION, 4-93
  - SHOW USERS, 4-93
  - SHOW WORKING SET, 4-93
  - STOP, 4-94 to 4-96
  - STOP/ABORT, 4-97
  - STOP/ENTRY, 4-98
  - STOP/REQUEUE, 4-99
  - SUBMIT, 4-100 to 4-105
  - summary, 1-4 to 1-6
  - syntax difference between
    - DCL and MCR, 4-1
  - TIME, 4-106
  - TYPE, 4-107 to 4-108
  - UFD, 2-8, 4-109 to 4-110
- Command language,
  - available under VAX/VMS, 1-1
  - default login file, 3-1
  - default user authorization
- Command language (Cont.)
  - file entry, 3-1 to 3-2, 4-105
  - selection, 1-1 to 1-4, 3-2
- Command line,
  - abbreviation, 2-19
  - comment placement on, 2-19
  - continuation, 2-19
  - keyword placement, 2-20
- Command procedure,
  - activation, 4-41, 4-100
- Comment,
  - example, 2-19
  - exclamation mark (!), 2-19
  - placement on command line, 2-19
  - semicolon (;), 2-19
- Comment for indirect command file, 5-2
- Component,
  - See Utility
- Concealed device, 2-16
- Conditional indirect file directive, 5-5
- CONNECT directive, 1-8
- CONNECT TO INTERRUPT VECTOR directive, 1-8
- Continuation,
  - command line, 2-19
- CONTINUE command, 4-12 to 4-13, 5-12
- Continue processing, 5-3
- Control character,
  - echo, 2-17
  - key, 2-17
- Controller designation, 2-8
- COPY command, 4-14 to 4-17
  
- .DATA indirect command file directive, 5-3
- Data recognition indirect command file directive, 5-11 to 5-12
- Date,
  - See Time
- <DATE> special symbol, 5-9
- DBG\$INPUT device, 2-14
- DBG\$OUTPUT device, 2-14
- DCL (DIGITAL Command Language), 1-1, 1-7
- /DE switch, 5-8
- DEALLOCATE command, 4-18
- DEBUG command, 3-7, 4-19
- .DEC indirect command file directive, 5-3
- DECnet-VAX, 4-89

## INDEX

- Decrement symbol value
  - indirect command file directive, 5-3
- <DEFAULT> special symbol, 5-9
- Delay indirect command file directive, 5-12
- .DELAY indirect command file directive, 5-3, 5-12
- DELETE command, 4-20 to 4-22
- Delete file after processing switch, 5-8
- DELETE/ENTRY command, 4-23 to 4-24
- Delta time,
  - example, 2-23
  - specification, 2-23
- DEPOSIT command, 4-25 to 4-27
- Device,
  - See also Pseudodevice
  - allocation, 3-2 to 3-3, 4-2
  - DBG\$INPUT, 2-14
  - DBG\$OUTPUT, 2-14
  - deallocation, 4-18
  - dismount, 4-36
  - generic name, 3-3
  - initialization, 4-45
  - logical name, 2-12, 3-4 to 3-5
  - mapping from RSX-11M to VAX/VMS device name, 2-15 to 2-16
  - mnemonic, 2-7
  - mount, 3-3 to 3-4, 4-55
  - name, 2-6 to 2-8
  - rooted, 2-17
  - SYSS\$COMMAND, 2-13
  - SYSS\$DISK, 2-13 to 2-14
  - SYSS\$ERROR, 2-13
  - SYSS\$ERRORLOG, 2-14
  - SYSS\$EXAMPLE, 2-14
  - SYSS\$HELP, 2-14
  - SYSS\$INPUT, 2-13, 5-11
  - SYSS\$INSTRUCTION, 2-14
  - SYSS\$LIBRARY, 2-14
  - SYSS\$LOGIN, 2-14
  - SYSS\$MAINTENANCE, 2-14
  - SYSS\$MANAGER, 2-14
  - SYSS\$MESSAGE, 2-14
  - SYSS\$NET, 2-14
  - SYSS\$NODE, 2-14
  - SYSS\$OUTPUT, 2-13, 5-11
  - SYSS\$SCRATCH, 2-14
  - SYSS\$SHARE, 2-14
  - SYSS\$SYSDEVICE, 2-14
  - SYSS\$SYSROOT, 2-15, 2-17
  - SYSS\$SYSTEM, 2-15
  - SYSS\$TEST, 2-15
  - SYSS\$UPDATE, 2-15
  - TT, 2-14
- Device mnemonic,
  - special symbol for system device, 5-9
- Device name,
  - controller designation, 2-8
  - default, 2-9
  - device mnemonic, 2-7
  - part of file specification, 2-6
  - unit number, 2-8
- Device type,
  - mnemonic list, 2-7
- DIGITAL Command Language (DCL),
  - See DCL (DIGITAL Command Language)
- Directive,
  - See also Indirect command file directive
  - added since RSX-11M Version 3.2, 1-8
  - CONNECT, 1-8
  - CONNECT TO INTERRUPT VECTOR, 1-8
  - not supported under VAX/VMS, 1-8
- Directory,
  - creation, 4-109
  - delimiter use in directory name, 2-6
  - for transportability, 1-8
  - information, 4-28
  - list, 4-28
  - MFD (Master File Directory), 2-17
  - name, 2-8
  - root, 2-17
  - special symbol for current, 5-9
  - UIC format, 1-8
- DIRECTORY command, 4-28 to 4-35
- Directory name,
  - default, 2-9
  - part of file specification, 2-6
  - subdirectory, 2-6, 2-8
  - UIC format, 2-6, 2-8
- .DISABLE DATA indirect command file directive, 5-3
- .DISABLE DOLLARS indirect command file directive, 5-3
- .DISABLE ESCAPE indirect command file directive, 5-3
- .DISABLE GLOBAL indirect command file directive, 5-3

## INDEX

- .DISABLE LOWERCASE indirect command file directive, 5-3
- .DISABLE QUIET indirect command file directive, 5-3
- .DISABLE SUBSTITUTION indirect command file directive, 5-3
- Disk Save and Compress Utility for Files-11 Structure Level 1, 1-3
- Disk Save and Compress Utility for Files-11 Structure Level 2, 1-4, 3-10
- DMO command, 4-36 to 4-37
- Dollar sign recognition indirect command file directive, 5-11 to 5-12
  
- EDI editor, 1-3, 3-8
- Editor,
  - EDI, 1-3, 3-8
  - EDT, 1-4, 3-10
  - SOS, 1-4, 3-10
- EDT editor, 1-4, 3-10
- .ENABLE DATA indirect command file directive, 5-4
- .ENABLE DOLLARS indirect command file directive, 5-4, 5-11 to 5-12
- .ENABLE ESCAPE indirect command file directive, 5-4
- .ENABLE GLOBAL indirect command file directive, 5-4
- .ENABLE LOWERCASE indirect command file directive, 5-4
- .ENABLE QUIET indirect command file directive, 5-4
- .ENABLE SUBSTITUTION indirect command file directive, 5-4
- .END indirect command file directive, 5-4
- .ERASE GLOBAL indirect command file directive, 5-4
- .ERASE LOCAL indirect command file directive, 5-4
- Error,
  - message, 2-24
- Escape character recognition,
  - disable, 5-3
  - enable, 5-4
- <ESCAPE> special symbol, 5-9
  
- EXAMINE command, 4-25, 4-38 to 4-40
- Execute procedure, 4-41 to 4-42
- @ (Execute procedure), 4-41 to 4-42
- Execution,
  - example with ODT, 4-76
  - interrupt, 2-18
  - native image, 1-1, 3-2, 3-6, 4-76 to 4-77
  - resume, 4-12
  - suspend, 5-6, 5-12
  - task image, 1-1, 3-2, 3-6, 4-76 to 4-77
- Execution of task image, requirement list under VAX/VMS, 1-2
- Existence of file,
  - test, 5-6
- EXIT command, 4-43 to 4-44, 4-66
- .EXIT indirect command file directive, 5-4
- <EXSTAT> special symbol, 4-44, 5-9
  
- 'F\$DIRECTORY lexical function, 5-14
- 'F\$EXTRACT lexical function, 5-14
- 'F\$LENGTH lexical function, 5-14
- 'F\$LOCATE lexical function, 5-14
- 'F\$LOGICAL lexical function, 5-14
- 'F\$MESSAGE lexical function, 5-14
- 'F\$MODE lexical function, 5-14
- 'F\$PROCESS lexical function, 5-14
- 'F\$TIME lexical function, 5-14
- 'F\$USER lexical function, 5-14
- 'F\$VERIFY lexical function, 5-14
- FCS (File Control Services), 1-8
- <FILATR> special symbol, 5-6
- File,
  - append, 4-4
  - copy, 4-14
  - create directory, 4-109
  - deletion, 4-20, 4-72
  - information, 4-28
  - list, 4-28
  - print, 4-68

## INDEX

- File (Cont.)
  - protection, 2-4
  - rename, 4-74
  - RSX-11M, 1-3
  - specification, 2-5
  - view, 4-107
- File deletion after processing switch, 5-8
- File Dump Utility (DMP), 1-3
- File name,
  - default, 2-9
  - part of file specification, 2-6
- File Patch Utility (PAT), 1-3
- File protection,
  - DELETE access, 2-21
  - EXECUTE access, 2-21
  - GROUP user category, 2-20
  - OWNER user category, 2-20
  - READ access, 2-21
  - specification of, 2-20 to 2-21
  - SYSTEM user category, 2-20
  - WORLD user category, 2-20
  - WRITE access, 2-21
- File specification,
  - default, 2-9
  - device name, 2-6
  - directory name, 2-6
  - file name, 2-6
  - file type, 2-6
  - file version number, 2-6
  - format, 2-5
  - node name, 2-6
  - wild card character, 2-7
- File Transfer Program (FLX), 1-3, 3-9
- File type,
  - default, 2-9 to 2-12
  - part of file specification, 2-6
- File version number,
  - default, 2-9
  - negative value use, 2-8
  - part of file specification, 2-6
- <FILERR> special symbol, 5-6
- Files-11, 1-7
- <FILSPC> special symbol, 5-6
  
- GET SENSE SWITCH directive, 1-8
- Global symbol definition,
  - disable, 5-3
  - enable, 5-4
  - erase, 5-4
- Global symbol table, 5-14
  
- .GOSUB indirect command file directive, 5-4
- .GOTO indirect command file directive, 5-4
- Group logical name table, 2-13
  
- HELLO command,
  - equivalent function in VAX/VMS MCR, 4-52
- Hibernation, 4-85
  
- .IF indirect command file directive, 5-5
- .IF[N]ACT indirect command file directive, 5-5, 5-7
- .IF[N]DF indirect command file directive, 5-5
- .IF[N]INS indirect command file directive, 5-5, 5-7
- .IF[N]LOA indirect command file directive, 5-5
- .IF[N]OLA indirect command file directive, 5-7
- .IFF indirect command file directive, 5-5
- .IFT indirect command file directive, 5-5
- Image,
  - See Native image
  - See Task image
- .INC indirect command file directive, 5-5
- Increment symbol value indirect command file directive, 5-5
- Indirect command file directive,
  - added since RSX-11M Version 3.2, 5-8
- .ASK, 5-2
- .ASKN, 5-2, 5-10 to 5-11
- .ASKS, 5-2, 5-15
- .BEGIN, 5-2
- .CHAIN, 5-3
- .CLOSE, 5-3
- .DATA, 5-3
- .DEC, 5-3
- .DELAY, 5-3, 5-12
- .DISABLE DATA, 5-3
- .DISABLE DOLLARS, 5-3
- .DISABLE ESCAPE, 5-3
- .DISABLE GLOBAL, 5-3
- .DISABLE LOWERCASE, 5-3
- .DISABLE QUIET, 5-3
- .DISABLE SUBSTITUTION, 5-3

## INDEX

- Indirect command file
  - (Cont.)
  - .ENABLE DATA, 5-4
  - .ENABLE DOLLARS, 5-4, 5-11 to 5-12
  - .ENABLE ESCAPE, 5-4
  - .ENABLE GLOBAL, 5-4
  - .ENABLE LOWERCASE, 5-4
  - .ENABLE QUIET, 5-4
  - .ENABLE SUBSTITUTION, 5-4
  - .END, 5-4
  - .ERASE GLOBAL, 5-4
  - .ERASE LOCAL, 5-4
  - .EXIT, 5-4
  - .GOSUB, 5-4
  - .GOTO, 5-4
  - .IF, 5-5
  - .IF[N]ACT, 5-5, 5-7
  - .IF[N]DF, 5-5
  - .IF[N]INS, 5-5, 5-7
  - .IF[N]LOA, 5-5, 5-7
  - .IFF, 5-5
  - .IFT, 5-5
  - .INC, 5-5
  - .label, 5-2
  - not defined, 5-7
  - .ONERR, 5-5, 5-7 to 5-8
  - .OPEN, 5-5
  - .OPENA, 5-5
  - .OPENR, 5-6
  - .PARSE, 5-6
  - .PAUSE, 5-6, 5-12
  - .READ, 5-6
  - .RETURN, 5-6
  - .SETF, 5-6
  - .SETN, 5-6, 5-10
  - .SETS, 5-6
  - .SETT, 5-6
  - slash (/), 5-2
  - .STOP, 5-6
  - .TEST, 5-6
  - .TESTFILE, 5-6
  - !text or ;text comment, 5-2
  - unsupported, 5-7 to 5-8
  - .WAIT, 5-7 to 5-8
  - .XQT, 5-7 to 5-8
- Indirect command file,
  - activation, 3-10, 4-41, 4-100, 4-105, 5-7
  - directive,
    - See Indirect command file directive
  - directive summary, 5-2 to 5-6
  - error condition detection, 4-66
  - interactive use at the terminal, 5-1
  - interruption, 4-95
- Indirect command file
  - (Cont.)
  - label table entry, 5-10
  - lexical function support, 5-13 to 5-14
  - nested, 4-105, 5-13
  - numeric symbol support, 5-10
  - parameter specification, 4-41
  - RSX-11M Version 3.2
    - directive support, 5-1
  - special symbol,
    - See Special symbol
  - special symbol support, 5-8 to 5-9
  - support under VAX/VMS, 5-14
  - switch support, 5-8
  - termination, 4-43, 4-94 to 4-96, 5-15
  - termination, effect on global symbol table, 5-14
  - termination, effect on an open data file, 5-14
  - termination indirect command file directive, 5-6
  - tracing of, 5-8
  - use of CTRL/Z, 5-15
  - use of ON command, 4-43, 4-67
- INITIALIZE command, 4-45 to 4-51
- Keyboard,
  - function key list, 2-17 to 2-18
- Label branch indirect command file directive, 5-4 to 5-5
- Label definition indirect command file directive, 5-2
- .Label indirect command file directive, 5-2
- Label table, 5-10
- LB pseudodevice, 2-15 to 2-17
- Length of string symbol, test, 5-6
- Lexical function,
  - 'F\$DIRECTORY, 5-14
  - 'F\$EXTRACT, 5-14
  - 'F\$LENGTH, 5-14
  - 'F\$LOCATE, 5-14
  - 'F\$LOGICAL, 5-14
  - 'F\$MESSAGE, 5-14

## INDEX

- Lexical function (Cont.)
  - 'F\$MODE, 5-14
  - 'F\$PROCESS, 5-14
  - 'F\$TIME, 5-14
  - 'F\$USER, 5-14
  - 'F\$VERIFY, 5-14
  - support for indirect command file use, 5-13 to 5-14
- Librarian (LBR), 1-3
- Library,
  - RSX-11M, 1-3
- <LIBUIC> special symbol, 5-9
- Limits and resource quotas,
  - B-1
- Local symbol definition,
  - erase, 5-4
- Logical name,
  - assignment, 2-12, 3-4 to 3-5, 4-7
  - concealed device, 2-16
  - deletion, 4-7
  - device, 2-16
  - difference from RSX-11M, 2-12
  - for RSX-11M task image, 3-4
  - for VAX-11 native image, 3-5
  - group logical name table, 2-13
  - process logical name table, 2-12 to 2-14
  - system logical name table, 2-13 to 2-15
  - system-defined, 2-13 to 2-15
  - translation, 2-12
- Login procedure, 3-1, 4-52 to 4-53
- LOGOUT command, 4-9, 4-54, 4-89
- Lowercase character,
  - enable conversion to uppercase, 5-3
  - enable use in strings, 5-4
  
- MACRO-11 Assembler (MAC), 1-3, 3-8
- Magnetic tape,
  - ANSI (17-character) file name support, 1-8
  - use of ANSI-labeled tape volume, 3-9
  - use of DOS-11 tape volume, 3-9
  - use of RT-11 tape volume, 3-9
- MAIL, 1-4, 3-10
- <MAPPED> special symbol, 5-9
  
- Mapping,
  - RSX-11M physical device name, 2-16
  - RSX-11M pseudodevice name, 2-15
  - /MC switch, 5-8
- MCR (Monitor Console Routine),
  - See also Command
  - See also Indirect command file directive
  - command language, 1-1
- Memory size special symbol, 5-9
- <MEMSIZ> special symbol, 5-9
- Message,
  - error, 2-24
  - format, 2-24
- MFD (Master File Directory), 2-17
- Monitor Console Routine (MCR),
  - See MCR (Monitor Console Routine)
- MOUNT command, 3-3 to 3-4, 4-55 to 4-65
  
- Name,
  - concealed device, 2-16
  - directory, 2-6, 2-8
  - file, 2-6
  - logical, 2-15, 4-7
  - logical device, 2-12 to 2-14, 2-16
  - node, 2-6
  - physical device, 2-6 to 2-8, 2-16
  - process, 2-2
  - pseudodevice, 2-15 to 2-16
  - RSX-11M file, 1-3
  - RSX-11M library, 1-3
  - RSX-11M utility, 1-3
  - VAX/VMS utility, 3-10
- Native image,
  - execution, 1-1, 3-2, 3-6, 4-76 to 4-77
  - logical name assignment, 3-5
  - program development, 1-1
  - termination, 4-94
- NETgen for RSX-11S or RSX-11M system, 5-1
- Node name,
  - default, 2-9
  - part of file specification, 2-6
  - /NODE switch, 5-8
  - /NOMC switch, 5-8

## INDEX

- Nonprivileged task library UIC
  - special symbol, 5-9
  - /NOTR switch, 5-8
- Numeric symbol, 5-10
  
- ODT, 1-3, 3-7, 4-19
- ODT.OBJ, 1-3
- ON severity-level statement, 4-66 to 4-67
- .ONERR indirect command file directive, 5-5, 5-7 to 5-8
- Open file to be read indirect command file directive, 5-6
- .OPEN indirect command file directive, 5-5
- .OPENA indirect command file directive, 5-5
- .OPENR indirect command file directive, 5-6
- OV pseudodevice, 2-15
  
- .PARSE indirect command file directive, 5-6
- .PAUSE indirect command file directive, 5-6, 5-12
- Peripheral Interchange Program (PIP), 1-3, 3-9
  - difference between PIP and PURGE, 4-73
  - /UF switch support, 3-8, 4-110
- PLAS directive, 1-8
- PRINT command, 4-68 to 4-71
- PRINT Utility,
  - equivalent function in VAX/VMS MCR, 4-71
- Privilege, 2-5, A-1
- Process,
  - and image, 2-1
  - characteristic, 2-1 to 2-2
  - comparison to a task, 2-1
  - detached, 2-2
  - identification, 2-2 to 2-3
  - logical name table, 2-12
  - login, 2-2
  - name, 2-2
  - priority, 2-5
  - privilege, 2-5, A-1
  - protection, 2-4
  - resource quotas and limits, 4-85, B-1
  - subprocess, 2-2
- Process logical name table, 2-12 to 2-13
  
- Program development, 1-2
  - native image, 1-1
  - task image, 1-1 to 1-2, 3-7, 3-11 to 3-12
  - tool, 1-2
- Protection, 2-3
  - file, 2-4
  - process, 2-4
- Pseudodevice,
  - CL, 2-15 to 2-16
  - CO, 2-15 to 2-16
  - LB, 2-15 to 2-17
  - name mapping, 2-15
  - OV, 2-15
  - SP, 2-15
  - SY, 2-15, 4-8
  - TI, 2-15 to 2-16
  - WK, 2-15
- PURGE command, 4-72 to 4-73
  
- Queue,
  - abort current batch job, 4-98
  - abort current print job, 4-97
  - activate batch job, 4-100
  - delete job, 4-20, 4-23
  - stop and requeue current print job, 4-99
  
- <RAD50> special symbol, 5-9
- Radix indicator, 5-10
- .READ indirect command file directive, 5-6
- Read record indirect command file directive, 5-6
- RECEIVE DATA directive, 3-12
- RECEIVE DATA OR EXIT directive, 3-12
- Record locking in compatibility mode, 1-8
- RENAME command, 4-74 to 4-75
- Requirement list,
  - for task image execution for compatibility mode, 1-2
- Resource quotas and limits, 4-85, B-1
- Return code, 4-43 to 4-44, 4-66 to 4-67
- Return from subroutine, 5-6
- .RETURN indirect command file directive, 5-6
- RMS-11,
  - utility, 1-3

## INDEX

- RMS-11 (Record Management System for the PDP-11), 1-8
- Root directory, 2-17
- Rooted device, 2-17
- <RSX11D> special symbol, 5-9
- RSX-11M,
  - file, 1-3
  - library, 1-3
- RSX-11M utility,
  - available under DCL, 1-3
  - available under MCR, 1-3
  - name to invoke, 1-3
- RSX-11S system generation
  - requirement list, 5-15 to 5-16
- RSXMAC.SML, 1-3
- RUN (Image) command, 4-76
- RUN (Process) command, 4-77 to 4-86
  
- Secondary file,
  - close, 5-3
  - open, 5-5
  - write data to, 5-3
- SET ACCOUNTING command, 4-87
- SET CARD READER command, 4-87
- SET DAY command, 4-87
- SET DEFAULT command, 4-87
- SET DEVICE command, 4-87
- SET DIRECTORY command, 4-87
- SET HOST command, 4-89
- SET LOGINS command, 4-87
- SET MAGTAPE command, 4-88
- SET MESSAGE command, 4-88
- SET [NO]CONTROL\_Y command, 4-87
- SET [NO]ON command, 4-67, 4-88
- SET [NO]VERIFY command, 4-88
- SET ON command, 4-66
- SET PASSWORD command, 4-90
- SET PRINTER command, 4-88
- SET PROCESS command, 4-88
- SET PROCESS/PRIORITY command, 4-88
- SET PROTECTION command, 2-21, 4-88
- SET PROTECTION/DEFAULT command, 4-88
- SET PROTECTION/DEVICE command, 4-87
- SET QUEUE/ENTRY command, 4-88
- SET RMS DEFAULT command, 4-88
- Set symbol value indirect command file directive, 5-6
- SET TERMINAL command, 4-88
- SET TIME command, 4-88
- SET /UIC command, 4-91
- SET VOLUME command, 4-88
- SET WORKING SET command, 4-88
- .SETF indirect command file directive, 5-6
- .SETN indirect command file directive, 5-6, 5-10
- .SETS indirect command file directive, 5-6
- .SETT indirect command file directive, 5-6
- SHOW [DAY]TIME command, 4-92
- SHOW DEFAULT command, 4-92
- SHOW DEVICES command, 4-92
- SHOW ERROR command, 4-92
- SHOW LOGICAL command, 2-12, 4-92
- SHOW MAGTAPE command, 4-92
- SHOW MEMORY command, 4-92
- SHOW NETWORK command, 4-92
- SHOW PRINTER command, 4-92
- SHOW PROCESS command, 2-3, 4-92
- SHOW PROTECTION command, 4-92
- SHOW QUEUE command, 4-92
- SHOW QUOTA command, 4-92
- SHOW RMS DEFAULT command, 4-93
- SHOW STATUS command, 4-93
- SHOW SYMBOL command, 4-93
- SHOW SYSTEM command, 2-3, 4-93
- SHOW TERMINAL command, 4-93
- SHOW TRANSLATION command, 4-93
- SHOW USERS command, 4-93
- SHOW WORKING SET command, 4-93
- Slash (/) indirect command file directive, 5-2
- SOS editor, 1-4, 3-10
- SP pseudodevice, 2-15
- Special symbol,
  - <ALPHAN>, 5-9
  - <CLI>, 5-9
  - <DATE>, 5-9
  - <DEFAULT>, 5-9
  - <ESCAPE>, 5-9
  - <EXSTAT>, 4-44, 5-9
  - <FILATR>, 5-6
  - <FILERR>, 5-6
  - <FILSPC>, 5-6
  - <LIBUIC>, 5-9
  - <MAPPED>, 5-9
  - <MEMSIZ>, 5-9
  - <RAD50>, 5-9
  - <RSX11D>, 5-9
  - <STRLEN>, 5-9
  - <SYDISK>, 5-9
  - <SYSTEM>, 5-9
  - <SYSUIC>, 5-9
  - <SYUNIT>, 5-9

## INDEX

- Special symbol (Cont.)
  - <TIME>, 5-9
  - <UIC>, 5-9
- Specification,
  - absolute time, 2-21 to 2-23
  - combination absolute and delta time, 2-24
  - delta time, 2-23
  - file protection, 2-20 to 2-21
- STOP command, 4-94 to 4-96
- .STOP indirect command file
  - directive, 5-6
- STOP/ABORT command, 4-97
- STOP/ENTRY command, 4-98
- STOP/REQUEUE command, 4-99
- String to substring parse
  - indirect command file directive, 5-6
- <STRLEN> special symbol, 5-9
- SUBMIT command, 4-100 to 4-105
- Subroutine,
  - call, 5-4
  - return from, 5-6
- Switch support for indirect command file use, 5-8
- SY pseudodevice, 2-15, 4-8
- <SYDISK> special symbol, 5-9
- Symbol substitution,
  - disable, 5-3
  - enable, 5-4
- Symbol table,
  - global, 5-14
- Symbol value radix, 5-10
- SYSSCOMMAND device, 2-13
- SYSSDISK device, 2-13 to 2-14
- SYSSERROR device, 2-13
- SYSSERRORLOG device, 2-14
- SYSSEXAMPLE device, 2-14
- SYSSHELP device, 2-14
- SYSSINPUT device, 2-13, 5-11
- SYSSINSTRUCTION device, 2-14
- SYSSLIBRARY device, 2-14
- SYSSLOGIN device, 2-14
- SYSSMAINTENANCE device, 2-14
- SYSSMANAGER device, 2-14
- SYSSMESSAGE device, 2-14
- SYSSNET device, 2-14
- SYSSNODE device, 2-14
- SYSSOUTPUT device, 2-13, 5-11
- SYSSSCRATCH device, 2-14
- SYSSSHARE device, 2-14
- SYSSSYSDEVICE device, 2-14
- SYSSSYSROOT device, 2-15, 2-17
- SYSSSYSTEM device, 2-15
- SYSSTEST device, 2-15
- SYSSUPDATE device, 2-15
- SYSLIB.OLB, 1-3
- System,
  - access activation (Login procedure), 3-1, 4-52
  - access termination, 4-9, 4-54
  - obtain current time, 4-106, 5-9
  - reset processor node using LOGOUT command, 4-89
  - set characteristic information, 4-87 to 4-88
  - set default information, 4-87 to 4-88
  - set new UIC and default directory, 4-91
  - set processor node, 4-89
  - set user password, 4-90
  - view characteristic information, 4-92 to 4-93
  - view default information, 4-92 to 4-93
- System generation,
  - for RSX-11S system, 5-15 to 5-16
- System logical name table,
  - 2-13 to 2-14
- System UIC special symbol, 5-9
- <SYSTEM> special symbol, 5-9
- <SYSUIC> special symbol, 5-9
- <SYUNIT> special symbol, 5-9
- Task Builder,
  - needed for an IAS or RSX-11D task image, 1-2
  - needed to provide logical name extensions for FCS and RMS-11, 1-2
- Task Builder (TKB), 1-3, 3-8, 3-11 to 3-13
  - ASG option, 3-12
  - TASK option, 3-12
- Task image, 2-1
  - example of execution with ODT, 4-76
  - execution, 1-1, 3-2, 3-6, 4-76 to 4-77
  - execution under RSX-11M, 1-4
  - installation, 3-5 to 3-6
  - logical name assignment, 3-4 to 3-5
  - program development, 1-2, 3-7, 3-11 to 3-12
  - program development, 1-1
  - record locking in

## INDEX

- Task image (Cont.)
  - compatibility mode, 1-8
  - requirement list for compatibility mode, 1-2
  - termination, 4-94
  - unsupported directive list, 1-8
- TASK option for Task Builder, 3-12
- Terminal,
  - control character, 2-17
  - function key list, 2-17 to 2-18
- .TEST indirect command file directive, 5-6
- .TESTFILE indirect command file directive, 5-6
- ;text indirect command file comment, 5-2
- !text indirect command file comment, 5-2
- TI pseudodevice, 2-15 to 2-16
- Time,
  - absolute, 2-21 to 2-23
  - combination absolute and delta, 2-24
  - delta, 2-23
- TIME command, 4-106
- <TIME> special symbol, 5-9
- /TR switch, 5-8
- Transportability, 1-7 to 1-8
- Truncation,
  - command name, 2-19
  - keyword, 2-19
- TT device, 2-14
- TYPE command, 4-107 to 4-108
  
- UFD command, 2-8, 4-109 to 4-110
- <UIC> special symbol, 5-9
- Unit number, 2-8
- Unit number of system device special symbol, 5-9
- User authorization file (UAF), 2-3
  
- User category for file protection, 2-20
- User identification code (UIC), 2-3 to 2-4
- Utility,
  - name to invoke RSX-11M, 1-3
  - name to invoke VAX/VMS, 3-10
  - RSX-11M, 1-3
  - VAX/VMS, 1-4, 3-10
  - VAX/VMS and RSX-11M differences, 3-10
- VAX-11 Symbolic Debugger (DEBUG command), 1-4, 3-10, 4-19
- VAX/VMS Personal Mail Utility, 1-4, 3-10
- VAX/VMS utility, available under MCR, 1-4, 3-10
- Virtual address, replacement, 4-25
- view, 4-38
- Volume,
  - dismount, 4-36
  - initialization, 4-45
  - mount, 3-3 to 3-4, 4-55
  
- .WAIT indirect command file directive, 5-7 to 5-8
- Wake-up request, cancel, 4-10
- Wild card character, use in file specification, 2-7
- WK pseudodevice, 2-15
  
- .XQT indirect command file directive, 5-7 to 5-8
  
- ZAP Utility (ZAP), 1-3



READER'S COMMENTS

NOTE: This form is for document comments only. DIGITAL will use comments submitted on this form at the company's discretion. If you require a written reply and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Did you find this manual understandable, usable, and well organized? Please make suggestions for improvement.

---

---

---

---

---

---

---

---

---

---

Did you find errors in this manual? If so, specify the error and the page number.

---

---

---

---

---

---

---

---

---

---

Please indicate the type of user/reader that you most nearly represent.

- Assembly language programmer
- Higher-level language programmer
- Occasional programmer (experienced)
- User with little programming experience
- Student programmer
- Other (please specify) \_\_\_\_\_

Name \_\_\_\_\_ Date \_\_\_\_\_

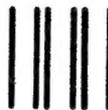
Organization \_\_\_\_\_

Street \_\_\_\_\_

City \_\_\_\_\_ State \_\_\_\_\_ Zip Code \_\_\_\_\_  
or Country

Do Not Tear - Fold Here and Tape

**digital**



No Postage  
Necessary  
if Mailed in the  
United States

**BUSINESS REPLY MAIL**  
FIRST CLASS PERMIT NO.33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE

BSSG PUBLICATIONS ZK1-3/J35  
DIGITAL EQUIPMENT CORPORATION  
110 SPIT BROOK ROAD  
NASHUA, NEW HAMPSHIRE 03061



Do Not Tear - Fold Here

Cut Along Dotted Line