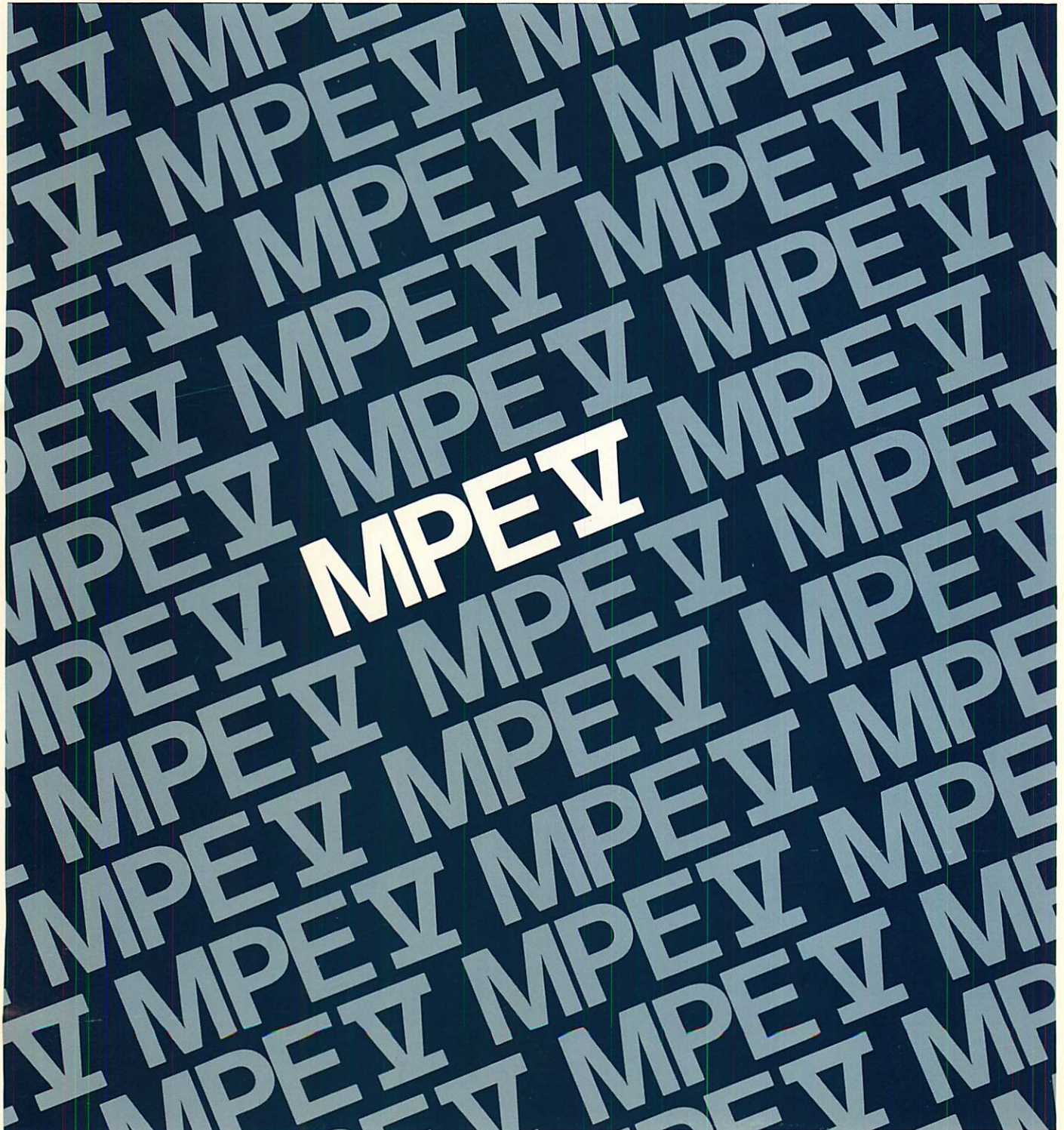


**MPE V Utilities Reference Manual
Reference Manual**



HP 3000 Computer Systems

**MPE V UTILITIES
Reference Manual**

Reference Manual



19447 PRUNERIDGE AVENUE, CUPERTINO, CA 95014

Part No. 32033-90008
(E0784)

Printed in U.S.A. 07/84

NOTICE

The information contained in this document is subject to change without notice.

HEWLETT-PACKARD MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance or use of this material.

Hewlett-Packard assumes no responsibility for the use or reliability of its software on equipment that is not furnished by Hewlett-Packard.

This document contains proprietary information which is protected by copyright. All rights are reserved. No part of this document may be photocopied, reproduced or translated to another language without the prior written consent of Hewlett-Packard Company.

LIST OF EFFECTIVE PAGES

The List of Effective Pages gives the date of the current edition, and lists the dates of all changed pages. Unchanged pages are listed as "ORIGINAL". Within the manual, any page changed since the last edition is indicated by printing the date the changes were made on the bottom of the page. Changes are marked with a vertical bar in the margin. If an update is incorporated when an edition is reprinted, these bars and dates remain. No information is incorporated into a reprinting unless it appears as a prior update.

First Edition July 1984

Effective Pages	Date
-----------------	------

ALL	ORIGINAL
---------------	----------

PRINTING HISTORY

New editions are complete revisions of the manual. Update packages, which are issued between editions, contain additional and replacement pages to be merged into the manual by the customer. The date on the title page and back cover of the manual changes only when a new edition is published. When an edition is reprinted, all the prior updates to the edition are incorporated. No information is incorporated into a reprinting unless it appears as a prior update.

The software date code number printed alongside the date indicates the version level of the software product at the time the manual edition or update was issued. Many product updates and fixes do not require manual changes, and conversely, manual corrections may be done without accompanying product changes. Therefore, do not expect a one-to-one correspondence between product updates and manual updates.

First Edition JUL 1984 E/F.00.00,
G.00.00

MPE V MANUAL PLAN

INTRODUCTORY LEVEL:

GENERAL
INFORMATION
Manual
5953-7553

GUIDE FOR THE
NEW USER
32033-90009
IN PROGRESS

GUIDE FOR THE
NEW OPERATOR
32033-90021
IN PROGRESS

STANDARD USER LEVEL:

MPE V COMMANDS
Reference
Manual
32033-90006

MPE V INTRINSICS
Reference
Manual
32033-90007

MPE V UTILITIES
Reference
Manual
32033-90008

SEGMENTER
Reference
Manual
30000-90011

DEBUG/STACK DUMP
Reference
Manual
30000-90012

FILE SYSTEM
Reference
Manual
30000-90236

ADMINISTRATIVE LEVEL:

MPE V SYSTEM OPERATION
& RESOURCE MANAGEMENT
Reference Manual
32033-90005

SUMMARY LEVEL:

MPE V
REFERENCE GUIDE
30000-90049
IN PROGRESS

There are many more manuals applicable to the HP 3000. A complete list may be found in every issue of the MPE V Communicator. Please contact your System Manager.

CONVENTIONS USED IN THIS MANUAL

NOTATION	DESCRIPTION
COMMAND	Commands are shown in CAPITAL LETTERS. The names must contain no blanks and be delimited by a non-alphabetic character (usually a blank).
KEYWORDS	Literal keywords, which are entered optionally but exactly as specified, appear in CAPITAL LETTERS .
<i>parameter</i>	Required parameters, for which you must substitute a value, appear in <i>bold italics</i> .
<i>parameter</i>	Optional parameters, for which you may substitute a value, appear in <i>standard italics</i> .
[]	<p>An element inside brackets is optional. Several elements stacked inside a pair of brackets means the user may select any one or none of these elements.</p> <p>Example: [A] [B] user may select A or B or neither.</p> <p>When brackets are nested, parameters in inner brackets can only be specified if parameters in outer brackets or comma place-holders are specified.</p> <p>Example: [parm1[,parm2[,parm3]]] may be entered as:</p> <p style="text-align: center;"><i>parm1,parm2,parm3</i> or <i>parm1,parm3</i> or <i>,,parm3</i> , etc.</p>
{ }	<p>When several elements are stacked within braces the user <i>must</i> select one of these elements.</p> <p>Example: { A } { B } user must select A or B.</p>
...	An ellipsis indicates that a previous bracketed element may be repeated, or that elements have been omitted.
<u>user input</u>	In examples of interactive dialog, user input is underlined. Example: NEW NAME? <u>ALPHA1</u>
superscript ^c	Control characters are indicated by a superscript ^c . Example: Y ^c (Press Y and the CNTL key simultaneously.)
<input type="text"/>	<input type="text"/> indicates a terminal key. The legend appears inside.
<<COMMENT>>	Programmer's comments in listings appear within << >> .

CONTENTS

Section	Page
PREFACE	xi

Section	Page
INTRODUCTION	
USER CAPABILITIES	1-1
UTILITIES COVERED IN THIS MANUAL	1-1

Section	Page
LISTEQ2/LISTEQ5	
OPERATION	2-1
EXAMPLE	2-2
LISTEQ2/LISTEQ5 ERROR MESSAGES	2-2

Section	Page
LISTDIR2/LISTDIR5	
OPERATION	3-1
COMMANDS	3-2
Security Restrictions Within LISTDIR5	3-2
Generic Names and 'Wild Card' Characters	3-3
Private Volumes	3-3
>LISTACCT	3-5
>LISTGROUP	3-6
>LISTUSER	3-7
>LISTF	3-8
>LISTSEC	3-10
>MOUNT	3-11
>DISMOUNT	3-12
>EXIT	3-13
>HELP	3-14
LISTDIR2/LISTDIR5 ERROR MESSAGES	3-15

Section	Page
SPOOK/SPOOK4/SPOOK5	
OPERATION	4-1
MPE V/P and MPE IV Systems	4-1
MPE V/E Systems	4-1
Multi-Version and Upgraded Sites	4-2
SPOOK Commands	4-2
Running SPOOK	4-3
Entering SPOOK Commands	4-3
Definition of Syntax for a Range List	4-4
Examples of Range Lists	4-4
>ALTER	4-5
>APPEND	4-7
>COPY	4-9
>DEBUG	4-11

CONTENTS (Continued)

>EXIT	4-12
>FIND	4-13
>HELP	4-15
>INPUT	4-16
>KILL	4-18
>LIST	4-19
>MODE	4-21
>OUTPUT	4-22
>PURGE	4-24
>QUIT	4-25
>RUN	4-26
>SHOW	4-28
>TEXT	4-30
>XPLAIN	4-31
SPOOK2/SPOOK5 WARNING AND ERROR MESSAGES	4-32
Warning Messages	4-32
Error Messages	4-32

Section	Page
FREE2/FREE5	
OPERATION	5-1
EXAMPLE	5-2
FREE2/FREE5 ERROR MESSAGES	5-2

Section	Page
MEMLOGAN	
OPERATION	6-1
MEMLOGAN Environment	6-2
Memory Error Log Formats	6-2
MEMLOGAN on Series 39/40/42/44/48	6-3
MEMLOGAN on Series 64 and 68	6-4
EXAMPLES	6-5
Series 39,40,42,44,48 MEMLOGAN Output	6-5
Series 64 and 68 MEMLOGAN Output	6-6
MEMLOGAN ERROR MESSAGES	6-6

Section	Page
MEMTIMER	
OPERATION	7-1
EXAMPLE	7-1
MMETIMER ERROR MESSAGES	7-2

Section	Page
LISTLOG2/LISTLOG5	
OPERATION	8-1
EXAMPLES	8-3
LISTLOG2/LISTLOG5 ERROR CONDITIONS	8-4

CONTENTS (Continued)

Section	Page
ASOCTABL/ASOCTBL5	
OPERATION	9-1
EXAMPLES	9-2
ASOCTABL/ASOCTBL5 ERROR MESSAGES	9-3

Section	Page
DPAN4/DPAN5	
OBTAINING A DPAN5 LISTING	10-1
Memory Dump on Series 39/40/42/44/48	10-2
Memory Dump on Series 64/68	10-3
DUMP ANALYSIS OPTIONS	10-3
OBTAINING A DPAN5 LISTING INTERACTIVELY	10-4
EXAMPLES	10-6
DPAN4/DPAN5 ERROR MESSAGES	10-8

Section	Page
SADUTIL	
PREPARING A SADUTIL COLD-LOAD MEDIUM	11-1
LOADING AND RUNNING SADUTIL	11-3
Cold-Loading the DUS on Series 39/40/42/44/48	11-3
Cold-Loading the DUS on Series 64/68	11-4
SADUTIL Configuration Dialogue	11-4
SADUTIL Device Types and Subtypes	11-7
CLID	11-8
CONF	11-9
COPY	11-10
EDIT	11-11
FIND	11-13
HELP	11-15
OUTM	11-16
PDSK	11-17
PDTT	11-19
PFIL	11-20
PVOL	11-22
SAVE	11-23
STOP	11-27
SADUTIL ERROR MESSAGES	11-28

Section	Page
RECOVER2/RECOVER5	
OPERATION	12-1
EXAMPLE	12-2
RECOVER2/RECOVER5 ERROR MESSAGES	12-3

CONTENTS (Continued)

Section	Page
DISKED2/DISKED5	
OPERATION	13-1
COMMANDS	13-2
>BASE	13-3
>DEBUG	13-4
>DISC	13-5
>DUMP	13-6
>EXIT	13-7
>FILE	13-8
>HELP	13-9
>LIST	13-10
>MODIFY	13-11
>WIDTH	13-12
EXAMPLES	13-13
DISKED2/DISKED5 ERROR MESSAGES	13-17

Section	Page
PATCH	
OPERATION	14-1
Using PMAP	14-2
Running PATCH	14-3
?D	14-5
?M	14-6
?DG	14-7
?MG	14-8
EXAMPLE	14-9
PATCH ERROR MESSAGES	14-9

Section	Page
SLPATCH	
RUNNING SLPATCH	15-1
?D	15-2
?M	15-3
EXAMPLE	15-4
SLPATCH ERROR MESSAGES	15-5

Section	Page
CAPABILITIES AND ACCESS MODES	
CAPABILITY SETS	A-1
User Attributes	A-1
File Attributes	A-1
Program Attributes	A-1
ACCESS MODE ABBREVIATIONS	A-2
USER-TYPE ABBREVIATIONS	A-2

PREFACE

This manual documents utility programs which are part of the Multi-Programming Executive (MPE) operating system running on the HP 3000 computer series.

This edition of the MPE V Utilities Reference Manual reflects the changes and enhancements to the internals of the MPE Operating System in the MPE V/E version (release G.00.00) and the MPE V/P version (E/F.00.00), and responses to service requests and reader comment sheets.

The INTRODUCTION is a new section in this edition. It was added to give you an overview of the programs included in the book; what their functions are, and what capabilities you must have to run them.

LISTDIR2/LISTDIR5 has new and extensive Error Message descriptions.

The SPOOK/SPOOK4/SPOOK5 introductory material is revised and now includes information on reading MPE IV and MPE V SPOOK tapes.

The descriptions of MEMTIMER, MEMLOGAN, FREE2/FREE5, LISTLOG, ASOCTABL/ASOCTBL5, and SLPATCH have been rewritten in friendlier formats.

The SADUTIL description now includes the instructions for creating the cold-load medium and loading SADUTIL, formerly found in appendices to the MPE IV version of this manual.

The DPAN4/DPAN5 has been extensively rewritten. The new material includes Examples text and a reorganization of the Dump Analysis Options.

DISKED2/DISKED5 includes new Examples material and additional Error Messages.

PATCH has new material to describe how to use PMAP to locate the memory location that you want to display or modify.

This manual is directed to users at many skill levels. It applies to those of you who know how to log on to the system and want some information on how much free disc space you have, to the System Manager who wants to know how the system resources are being allocated, and to the Programmer who is making on line changes to program code. The capability requirements to use each program are an indication, but not an infallible guide, to how complex the program is and the extent of its system impact.

If you need help or more information, the following supporting documentation provides in depth discussions of concepts used by the utility programs:

- MPE V INTRINSICS REFERENCE MANUAL (32033-90007)
- MPE V SYSTEM OPERATION and RESOURCE MANAGEMENT REFERENCE MANUAL (32033-90005)
- MPE V COMMANDS REFERENCE MANUAL (30000-90009)

We welcome your comments and suggestions on how we can make this manual more useful. Please use the postage-paid Reader Comment Sheets found at the back.

Utilities are programs of general usefulness, applicable to many jobs and purposes, which perform functions not available through MPE commands. Utilities are used in program development, file manipulation, and system administration to provide information access controls, report on system resources, and other special-purpose information. Certain utilities operate in a stand-alone mode, that is, without MPE running.

Many of the utilities in this manual have two versions, one for MPE V/E, and one for MPE V/P (and MPE IV). This is indicated in the section titles by a dual names, separated by a slash, as in "LISTEQ2/LISTEQ5". In all cases, the dialog and operation of the versions is the same. The only difference is the name of the file in which the program resides. Full details are presented in each section.

USER CAPABILITIES

The MPE Operating System is organized in such a way that each user of the system is assigned a capability set. The capability set can be unique to each user, or to each account, or to a group. It always, however, consists of the sum of User Attributes, File Attributes, and Program Attributes. Some functions within utility programs are limited by the capability set of the program's user. A brief description of the requirements of each is given below; more information is provided in the individual program descriptions. Finally, APPENDIX A contains an explanation of all of the User, File, and Program Attributes.

UTILITIES COVERED IN THIS MANUAL

The following utilities are used by general system users; some have special features available to users with non-standard capabilities:

- **LISTEQ2/LISTEQ5** lists the file equations and temporary files for the current session. No special capabilities are required.
- **LISTDIR2/LISTDIR5** lists the attributes of accounts, users, groups, and files. All users may list the attributes of their logon account, group, user name, and files (although some attributes are restricted). The Account Manager may list the attributes of other groups and users within the logon account, including passwords, lockwords, and creators. System Managers may list the attributes of any file, group, user, or account in the system.
- **SPOOK/SPOOK2/SPOOK5** displays, modifies and moves spooled device files. The extensive command set (eighteen functions) allows a variety of access levels. All users can access their own files, but cannot use the >INPUT or >OUTPUT commands. You must have Privilege Mode to use the DEBUG command. An Account Manager may access any spool file on the logon account, but cannot use the >INPUT or >OUTPUT commands, and must have Privileged Mode (PM) to use >DEBUG. The System Manager may access any spool file and use all commands, but must have Privileged Mode to use >DEBUG.

System Managers and other administrative users may use these utilities:

- **FREE2/FREE5** details the contiguous free space and total free space on discs, and the total free space in the system. No special capabilities are needed.
- **MEMLOGAN** prints out error logs recorded by the Memory Error Logging System from the memory error logging boards. System Manager capability is required.
- **MEMTIMER** sets the update intervals of the memory logging file used by **MEMLOGAN**. System Manager capability is required.
- **LISTLOG2/LISTLOG5** analyzes MPE system log files. It allows you to print events from a particular log file or files. System Manager capability is required.
- **ASOCTABL/ASOCTBL5** allows a System Manager to define which users are authorized to associate with which device classes. This utility can also be used to gain exclusive access to a device.

The following are used to recover from and analyze system problems:

- **DPAN4/DPAN5** produces a formatted memory dump listing from the dump taken by the Software Dump Facility. No special capabilities are required to run it.
- **SADUTIL** performs emergency disc operations after the system has gone down. This is a stand-alone program with no operating system controls. No special capability is required, but System Manager or System Supervisor is recommended.
- **RECOVER2/RECOVER5** reloads the files created by **SADUTIL** back into the system after a system failure. System Manager capability is required. This utility runs only in Privileged Mode within sessions.

Finally, the following utilities are often used by applications programmers and System Managers to modify disc files:

- **DISKED2/DISKED5** allows you to display and/or modify the sector contents of a disc, or segment contents of a file.
- **PATCH** displays and modifies object code of programs without recompiling. No special MPE capabilities are required to run **PATCH**.
- **SLPATCH** allows you to display or modify the contents of a Segmented Library file. **SLPATCH** runs in Privileged Mode, but does not require special capabilities.

LISTEQ5 lists the :FILE equations and temporary files in existence for the current session. No special MPE capabilities are required to run this program.

The program LISTEQ5.PUB.SYS is found only on MPE V/E systems; the MPE V/P version is LISTEQ2.PUB.SYS. The dialogue and operation of the two are identical. Throughout this section, LISTEQ5 will be used for all examples and explanations.

OPERATION

1. To run LISTEQ5, type:

```
:RUN LISTEQ5.PUB.SYS
```

2. LISTEQ5 identifies itself and outputs a "TEMP FILES" heading, followed by a list of temporary files on the system. This is followed by the heading "FILE EQUATIONS", with a list of file equations currently being used:

```
LISTEQ5 G.00.00 (C) HEWLETT-PACKARD CO., 1978
```

```
***TEMP FILES  
INPUT.PUB.TIMONS
```

```
***FILE EQUATIONS  
:FILE TAPE;DEV=TAPE  
:FILE PRINT;DEV=LP
```

3. By default, LISTEQ5 output goes to \$STDLIST, usually the terminal. The formal file designator is LIST. To redirect the output to the line printer (LP), use a file equation and run LISTEQ5 with ;PARAM=1, as follows:

```
:FILE LIST;DEV=LP  
:RUN LISTEQ5.PUB.SYS;PARAM=1
```


EXAMPLE

```
:FILE PRINT;DEV=LP  
:BUILD INPUT;REC=40,3,F,ASCII;TEMP  
  
:RUN LISTEQ5.PUB.SYS  
  
LISTEQ5 G.00.00 (C) HEWLETT-PACKARD CO., 1978  
  
***TEMP FILES  
INPUT.PUB.TIMONS  
  
***FILE EQUATIONS  
:FILE PRINT;DEV=LP  
  
END OF PROGRAM  
:
```

LISTEQ2/LISTEQ5 ERROR MESSAGES

FAILURE TO GET JDT	LISTEQ5 found a Job Directory Table of unexpected size. Be sure you have the current version of LISTEQ5.
FAILURE TO OPEN LISTFILE	The list file requested cannot be opened. Be sure the list file is specified correctly.
FAILURE TO WRITE ON LISTFILE	Check your file equations for device class names.

LISTDIR2/LISTDIR5

SECTION

III

LISTDIR5 provides information similar to, but more detailed than, the :LISTACCT, :LISTGROUP, and :LISTF MPE commands. Some information available from LISTDIR5 is only accessible to the System Manager or Account Manager.

The program LISTDIR5.PUB.SYS is found only on MPE V/E systems; the MPE V/P version is LISTDIR2.PUB.SYS. The dialogue and operation of the two are identical. Throughout this section, LISTDIR5 will be used for all examples and explanations.

The LISTDIR5 program allows you to:

- List the attributes of accounts, users, groups, and files.
- List the security provisions for one or more files.
- List the syntax rules for all LISTDIR5 commands.

OPERATION

1. To run LISTDIR5 enter:

```
:RUN LISTDIR5.PUB.SYS
```

2. LISTDIR5 identifies itself and prompts you for a command:

```
LISTDIR5 G.00.00 (C) HEWLETT-PACKARD CO., 1983  
TYPE 'HELP' FOR AID  
>
```

3. Enter one of the commands from the "LISTDIR5 COMMANDS" section.

When running LISTDIR5 in batch mode, commands must start in the first column of the input file. Interactively, there must be no space between the > prompt and the first character of the command.

Standard MPE security applies to LISTDIR5. Refer to "SECURITY RESTRICTIONS WITHIN LISTDIR5" for details.

If you do not have sufficient capability, restricted information is replaced by double asterisks (**). This includes passwords, lockwords, disc addresses, and creator identities.

4. Press Y^C to halt the execution of a LISTDIR5 command. To suspend LISTDIR5 when a command is not executing, press **(BREAK)**. You will be prompted for another command.

LISTDIR 2/LISTDIR 5

By default, LISTDIR5 dialogue takes place on the terminal. The formal file designator of the output file is OUT. To redirect the output, use a file equation, and use PARM=1 in your :RUN command. For example:

```
:FILE OUT; DEV=LP  
:RUN LISTDIR5.PUB.SYS; PARM=1
```

COMMANDS

The LISTDIR5 commands listed below are described in the following pages:

LISTACCT	Lists the characteristics of an account.
LISTGROUP	Lists the characteristics of a group.
LISTUSER	Lists the characteristics of a user.
LISTF	Lists the attributes of a file or file set.
LISTSEC	Lists the security provisions and capabilities of a user.
MOUNT	Generates a request at the System Console to mount a private volume.
DISMOUNT	Generates a request at the System Console to dismount a private volume.
EXIT	Terminates LISTDIR5.
HELP	Accesses the LISTDIR5 HELP facility.

Security Restrictions Within LISTDIR5

To prevent disclosure of group and account names and passwords, LISTDIR5 operates according to MPE security. The following rules apply:

- A System Manager has unrestricted access to the LISTDIR5 output information.
- An Account Manager may specify any group or user name in his logon account.
- A standard user (one who is neither a System Manager nor an Account Manager) may specify only his logon account, group, and user name.
- Passwords, lockwords, creator identities, file label addresses, and privileged file codes are displayed only when ;PASS is specified by qualified users.
- Account passwords may be listed only by System Managers.
- Group and user passwords may be listed only by an Account Manager or System Manager.
- File lockwords and creator names can be listed by an Account Manager or System Manager only.

- Disc file addresses and extent maps may be displayed by the creator of the file, or an Account Manager or System Manager.
- Privileged file codes may be displayed by an Account Manager, System Manager, or the creator of the file if the creator has Privileged Mode capability.
- A group name containing "wild card" characters may be used only by an Account Manager or System Manager (refer to "GENERIC NAMES AND 'WILD CARD' CHARACTERS" in this section).
- An account name containing "wild card" characters may be specified only by a System Manager.

Generic Names and 'Wild Card' Characters

File names, user names, group names, and account names can be input as generic names. This lets you request information on all items that meet a set of criteria, providing you are allowed access to the information (refer to "LISTDIR5 COMMANDS" in this section).

A generic name consists of up to eight alphanumeric and "wild card" characters. Wild characters indicate nonspecific alphanumeric characters, so a generic name can represent several actual names. The "wild card" characters are:

- # which represents exactly one numeric character,
- ? which represents exactly one alphanumeric character, and
- @ which represents zero or more alphanumeric characters.

A generic name must start with an alphabetic character, @, or ?. Examples of generic names are:

LISTF K#####.@	Lists all files starting with K and followed by seven digits, in any group (e.g. K1431254, K1418621).
LISTF L@X	Lists all files starting with L and ending with X, with any characters in between (e.g. LAX, LUMMX).
LISTF ?X@	Lists all files whose second letter is X (e.g. EXTRA, MX4).
LISTF @	Lists all files in the logon group.
LISTF @.PUB.@	Lists all files in the PUB group of all accounts (requires System Manager (SM) capability).

Private Volumes

You can use the >LISTF and >LISTSEC commands to list the attributes of files on private volumes, if you have Use Volume (UV) capability. If you do not have UV capability, or if the volume class/set is not physically mounted, appropriate warnings are issued.

LISTDIR2/LISTDIR5

LISTDIR5 can list entries in the Private Volume Directory even if there are no corresponding entries in the System Directory. The >MOUNT command must be used first to specify the volume set/class to be mounted. Again, you must have UV capability. Subsequent >LISTACCT, >LISTGROUP, >LISTF, and >LISTSEC commands display the directory entry and file label information contained on the private volume set/class. The account level and group level security displayed by these commands does not reflect access security utilized by MPE. MPE security is always derived from the account and group entries in the System Directory. The >LISTUSER command always displays the entry in the System Directory since there is no corresponding entry in the Private Volume Directory.

Only one volume set or class at a time can be mounted by LISTDIR5. Subsequent >MOUNT and >DISMOUNT commands cause the previously mounted volume set/class to be dismounted. The >DISMOUNT command redirects LISTDIR5 to the System Directory.

>LISTACCT

Lists attributes of an account or set of accounts.

SYNTAX

```
>LISTACCT [acctname] [,listfile]  
[ ;PASS]
```

PARAMETERS

- acctname* A one- to eight-character account name or generic name. Default is the logon account.
- listfile* Used to redirect the file or device which is to receive the output listing. It must be an existing temporary or permanent disc file, or, if preceded with "*", may back reference a file equation. Default is \$STDLIST.
- PASS Displays the password of the account.

EXAMPLE

```
>LISTACCT FUDD  
*****  
ACCOUNT: FUDD  
  
DISC SPACE; 4143(S)                    PASSWORD: **  
CPU TIME: 231 (SEC)                    LOC ATTR: %0  
CONNECT TIME: 232(MIN)                SECURITY--READ:    AC  
DISC LIMIT: UNLIMITED                WRITE:            AC  
CPU LIMIT: UNLIMITED                APPEND:           AC  
CONNECT LIMIT: UNLIMITED            LOCK:             AC  
MAX PRI: 150                          EXECUTE:          AC  
GRP INX PTR: %4346  
USR INX PTR: %4347  
CAP: AM,AL,GL,DI,OP,CV,UV,CS,ND,SF,IA,RA,PH,DS,MB,PM
```

>LISTGROUP

Lists attributes of an group or set of groups.

SYNTAX

```
>LISTGROUP [groupname [.acctname]] [,listfile]  
[ ;PASS]
```

PARAMETERS

- groupname* A one- to eight-character group name or generic name. Default is the logon group.
- acctname* A one- to eight-character account name or generic name. Default is the logon account.
- listfile* Used to redirect the file or device which is to receive the output listing. It must be an existing temporary or permanent disc file, or, if preceded with "*", may back reference a file equation. Default is \$STDLIST.
- PASS Displays the password of the group.

EXAMPLE

```
>LISTGROUP @.FUDD  
*****  
GROUP: PUB.FUDD  
  
DISC SPACE: 4143(S)                    PASSWORD: **  
CPU TIME: 231(SEC)                    SECURITY:--READ:        ANY  
CONNECT TIME: 232(MIN)                WRITE:                AD,GU  
DISC LIMIT: UNLIMITED                APPEND:                AL,GU  
CPU LIMIT: UNLIMITED                 LOCK:                 AL,GU  
CONNECT LIMIT: UNLIMITED             EXECUTE:              ANY  
FILE INX PTR: %4350                    SAVE:                 AL,GU  
MVTABX: %0                            PRIV VOL:             NO  
MOUNT REF CNT: 0  
HOME VOL SET:  
CAP: IA,BA,PH,DS,MR,PM
```

Lists attributes of a user or set of users.

SYNTAX

```
>LISTUSER [user[.acctname]] [,listfile]
[;PASS]
```

PARAMETERS

user A one- to eight-character user name or generic name. Default is the logon user.

acctname A one- to eight-character account name or generic name. Default is the logon account.

listfile Used to redirect the file or device which is to receive the output listing. It must be an existing temporary or permanent disc file, or, if preceded with "*", may back reference a file equation. Default is \$STDLIST.

PASS Displays the password of the user.

EXAMPLE

```
>LISTUSER ELMO;PASS
*****
USER: ELMO.FUDD

HOME GROUP: PUB                    PASSWORD: BUNNY
MAX PRI: 150                        LOC ATTR: %0
LOGON CNT: 1
CAP: AM,AL,GL,DI,OP,CV,UV,CS,ND,SF,IA,BA,PH,DS,MR,PM
```


>LISTF

Lists attributes of an file or set of files.

SYNTAX

```
>LISTF filename[.groupname[.acctname]] [,listfile]  
    [;PASS]  
    [;MAP]
```

PARAMETERS

<i>filename</i>	The file name or generic file set name to be listed.
<i>groupname</i>	The group name or generic group set name to be listed. Default is the logon group.
<i>acctname</i>	The account name or generic account set name to be listed. Default is the logon account.
<i>listfile</i>	Used to redirect the file or device which is to receive the output listing. It must be an existing temporary or permanent disc file, or, if preceded with "*", may back reference a file equation. Default is \$STDLIST.
PASS	Displays the lockword of the file.
MAP	Creates a list of the disc addresses of each file extent. You must be the files' creator, or have Account Manager (AM) or System Manager (SM) capability, to use this parameter.

EXAMPLE

```
>LISTF MYFILE;PASS;MAP
```

```
*****  
FILE: MYFILE.PUB.FUDD
```

```
FCODE: 0  
BLK FACTOR: 1  
REC SIZE: 256(B)  
BLK SIZE: 128(W)  
EXT SIZE: 128(S)  
# REC: 0  
# SEC: 128  
# EXT: 1  
MAX REC: 1023  
MAX EXT: 8  
# LABELS: 0  
MAX LABELS: 0  
DISC DEV #: 2  
DISC TYPE: 0  
DISC SUBTYPE: 9  
CLASS: DISC  
FCB VECTOR: %0  
EXT MAP: %301434601  
  
FOPTIONS: BINARY, FIXED, STD  
CREATOR: ELMO  
LOCKWORD: SECRET  
SECURITY--READ: ANY  
WRITE: ANY  
APPEND: ANY  
LOCK: ANY  
EXECUTE: ANY  
**SECURITY IS ON  
COLD LOAD ID: %14157  
CREATED: THU, 20 OCT 1983  
MODIFIED: THU, 20 OCT 1983  
ACCESSED: THU, 20 OCT 1983  
LABEL ADR: %1434601  
SEC OFFSET: %1  
FLAGS: NO ACCESSORS
```

The abbreviations "S", "B", and "W" stand for sectors, bytes, and words. "COLD LOAD ID" is a counter that keeps track of system cold loads, and helps identify the status of files when a dump is taken. "LABELS" and "MAX LABELS" refer to user written file labels, not standard labels written by MPE. "SEC OFFSET" indicates the number of sectors between the file label and the first data sector within the file. "FLAGS" indicates if and why a file has been locked down for exclusive access by a program. "FCB VECTOR" and "INX PTR" are useful to system analysts for debugging. "STD", "MSG", "CIR", and "KSAM" stand for standard, message, circular, and Keyed Sequential Access Method files, respectively.

>LISTSEC

Lists security provisions of an file or set of files.

SYNTAX

```
>LISTSEC filename[.groupname[.acctname]] [,listfile]  
[;PASS]
```

PARAMETERS

- filename* The file name or generic file set name to be listed.
- groupname* The group name or generic group set name to be listed. Default is the logon group.
- acctname* The account name or generic account set name to be listed. Default is the logon account.
- listfile* Used to redirect the file or device which is to receive the output listing. It must be an existing temporary or permanent disc file, or, if preceded with "*", may back reference a file equation. Default is \$STDLIST.
- PASS Displays the creator and lockword of the file.

EXAMPLE

```
>LISTSEC MYFILE  
*****  
FILE: MYFILE.PUB.FUDD  
  
SECURITY--READ: AC  
  (ACCT) WRITE: AC  
          APPEND: AC  
          LOCK: AC  
          EXECUTE: AC  
  
SECURITY--READ: ANY  
  (GROUP) WRITE: AL,GU  
          APPEND: AL,GU  
          LOCK: AL,GU  
          EXECUTE: ANY  
          SAVE: AL,GU  
  
SECURITY--READ: ANY                   FCODE: 0  
  (FILE) WRITE: ANY                   CREATOR: **  
          APPEND: ANY                 LOCKWORD: **  
          LOCK: ANY                   **SECURITY IS ON  
          EXECUTE: ANY
```

Generates a request at the System Console to mount a private volume.

SYNTAX

```
> MOUNT [ { vsname }  
         { * } [ .groupname [ .acctname ] ] ]  
  
         [ ;GEN=[genindex]
```

PARAMETERS

<i>vsname</i> or *	The volume set name to be listed; no generic volume set names are permitted. "*" indicates the home volume set (default).
<i>groupname</i>	The group name to be listed; you may only use a generic group name if you have Account Manager (AM) or System Manager (SM) capability. Default is the logon group.
<i>acctname</i>	The account name to be listed; a generic account name is not permitted. Default is the logon account.
<i>genindex</i>	The generation index of the volume set or class to be mounted. Range is 0 to 32767, or -1 (default), which indicates that any generation is permitted.

>DISMOUNT

Generates a request at the System Console to dismount the currently mounted volume set.

SYNTAX

```
>DISMOUNT
```

Terminates the LISTDIR 5 subsystem.

SYNTAX

>EXIT

>HELP

Accesses the LISTDIR5 HELP facility.

SYNTAX

```
>HELP
```

LISTDIR2/LISTDIR5 ERROR MESSAGES

INVALID COMMAND	Re-enter the command with the correct spelling and syntax.
NO SUCH FILE	An invalid file name was entered in the command. Check your spelling or do a >LISTF to see if this file is in your account.
NO SUCH GROUP	An invalid group name was specified in the command.
NO SUCH ACCOUNT	An invalid account name was specified in the command. Check the spelling and re-enter.
NO SUCH USER	An invalid user name was specified in the command.
INVALID PARAMETER	An invalid parameter was specified in the command.
DUPLICATE PARAMETER	The same parameter was specified twice in this command.
INVALID DESIGNATOR	An invalid designator was specified in the command.
ACCOUNT IS NOT LOG ON	You must have System Manager capability to gain information about an account other than the one you logged on to.
GROUP IS NOT LOG ON	You must have Account Manager capability to list information about groups other than your own.
USER IS NOT LOG ON	You must have Account Manager capabilities to access information about other users.
DISC I/O ERROR	An I/O error occurred while trying to access the directory.
MISSING PARAMETER	Check the parameter syntax and re-enter.
INVALID FILESET FOR USER	An invalid fileset for user files was specified in a command.
LOCKWORD IS NOT ALLOWED	A lockword specification is not allowed in a command.
CANNOT CLOSE LIST FILE--FSERR= <i>nnn</i>	An error was encountered in closing the list file. Check the FSERR number.
CANNOT OPEN LIST FILE--FSERR= <i>nnn</i>	An error was encountered in opening the list file. Check the FSERR number.
WRITE ERROR ON LIST FILE--FSERR= <i>nnn</i>	A write error was encountered on the list file.
INVALID LIST FILE NAME	Check the file name to be sure that it conforms to MPE naming conventions. If you are using a file equation, be sure that the device class in the equation can be used for listing, and that the list file name begins with a "*".
FILE NAME BEGINS WITH NUMERIC CHARACTER OR #	The first character of the file name must begin with a letter.

LISTDIR2/LISTDIR5

MISSING FILE NAME	The file name specification is missing from this command. Check the syntax and re-enter.
FILE NAME EXCEEDS 8 CHARACTERS	Check the file name and retry.
MISSING DELIMITER AFTER FILE NAME	Check the command syntax and re-enter.
GENERIC FILE NAME IS NOT ALLOWED	You cannot specify this file with a "wild card" character.
GROUP NAME BEGINS WITH NUMERIC CHARACTER OR #	The first character of this group name must be a letter.
MISSING GROUP NAME	Check the command syntax and re-enter.
GROUP NAME EXCEEDS 8 CHARACTERS	Change the group name and re-enter.
MISSING DELIMITER AFTER GROUP NAME	Check the command syntax and re-enter.
GENERIC GROUP NAME IS NOT ALLOWED	A group name containing "wild card" characters may be specified only by a System Manager or Account Manager.
ACCOUNT NAME BEGINS WITH NUMERIC CHARACTER OR #	The first character of the account name must be a letter.
MISSING ACCOUNT NAME	Check the command syntax and re-enter.
ACCOUNT NAME EXCEEDS 8 CHARACTERS	Check your spelling and re-enter the name.
MISSING DELIMITER AFTER ACCOUNT NAME	Check the command syntax and re-enter.
GENERIC ACCOUNT NAME IS NOT ALLOWED	An account name containing "wild card" characters may be specified only by a System Manager. No "wild card" characters are permitted in the <i>acctname</i> parameter of the >MOUNT command.
USER NAME BEGINS WITH NUMERIC CHARACTER OR #	The first character of the user name must be a letter. Correct the name and re-enter.
MISSING USER NAME	Check the command syntax and re-enter.
USER NAME EXCEEDS 8 CHARACTERS	Check your spelling and re-enter.
MISSING DELIMITER AFTER USER NAME	Check the syntax for the >LISTUSER command.
GENERIC USER NAME IS NOT ALLOWED	You may specify only your logon user name unless you are a System Manager or Account Manager.

VOLUME NAME BEGINS WITH NUMERIC CHARACTER OR #	The volume name must begin with a letter.
MISSING VOLUME NAME	Either a volume set name, or a back reference to the home volume set, is required in the >MOUNT command.
VOLUME NAME EXCEEDS 8 CHARACTERS	Check spelling and re-enter the name.
MISSING DELIMITER AFTER VOLUME NAME	Check the command syntax of the >MOUNT command.
GENERIC VOLUME NAME IS NOT ALLOWED	No "wild card" characters are allowed in the <i>vsname</i> parameter of the >MOUNT command.
USER DOES NOT HAVE UV CAPABILITY	You must have Use Volumes (UV) capability to access a private volume.
PRIVATE VOLUME ERROR= <i>nnn</i>	An error was encountered in accessing a private volume.
COMMAND IS NOT IMPLEMENTED	You have specified a command that is not functional in the current version of MPE.
INVALID GENERATION INDEX	The value you specified for <i>genindex</i> in the >MOUNT command must be between -1 and 32767.
>MOUNT REQUIRED BEFORE LISTING FILES ON PRIVATE VOLUME	You cannot use >LISTUSER, >LISTSEC, or >LISTF until after you have specified the volume set or class to be mounted with >MOUNT.
MPE COMMANDS MAY NOT BE EXECUTED FROM LISTDIR5	Enter <u>EXIT</u> before attempting any MPE commands.
ERROR FOUND ON INPUT. LISTDIR5 TERMINATED	An error was encountered in reading the input string.
EOF DETECTED. LISTDIR5 TERMINATED	A end-of-file condition was encountered in reading the input string.
EOF DETECTED ON OUTPUT FILE	An end-of-file condition was encountered in writing to the list file.

SPOOK/SPOOK4/SPOOK5

SECTION

IV

The SPOOK, SPOOK4, and SPOOK5 utility programs allow you to interrogate, manipulate, and transfer spooled device files (spool files) created and maintained by MPE. Any user can access this utility, but some functions are limited to users with Privileged Mode (PM), System Manager (SM), or Account Manager (AM) capabilities.

SPOOK is the version of the utility supplied with MPE IV and MPE V/P (E/F.00.00). Both SPOOK4 and SPOOK5 are supplied with MPE V/E. The command sets for all three versions are identical. If the System Manager has established a User-Defined Command (UDC) for the appropriate version of SPOOK, as described below, most users will be unaware of any changes. For the remainder of this section, the term "SPOOK" will be used as a generic reference to all three versions, unless indicated otherwise.

OPERATION

How you use SPOOK will depend on the operating system running on your computer.

MPE V/P and MPE IV Systems

If you are on MPE V/P (or on MPE IV), or if you have several systems running any combination of MPE V/P (E/F.00.00) and MPE IV, you will use the program SPOOK.PUB.SYS. The SPOOK tapes from both versions of MPE are completely compatible. To simplify the use of SPOOK, the System Manager could establish the following UDC:

```
SPOOK
COMMENT *** This UDC is for an MPE V/P or MPE IV system ***
RUN SPOOK.PUB.SYS
***
```

MPE V/E Systems

If you have one or more systems running only MPE V/E, you will use the program SPOOK5.PUB.SYS. SPOOK5 tapes may be read by any MPE V/E system. The System Manager might establish the following UDC:

```
SPOOK
COMMENT *** This UDC is for an MPE V/E system ***
RUN SPOOK5.PUB.SYS
***
```

Multi-Version and Upgraded Sites

If your site has systems running both MPE V/E, and MPE V/P (E/F.00.00) and/or MPE IV, you will need to use SPOOK5 on your MPE V/E system and SPOOK4 on your other system.

The creation of SPOOK4 and SPOOK5 was necessitated by the changes in system tables in MPE V/E. The format of the SPOOK tape includes system table entry images of the Device Class Table and the Output Device Directory, both of which underwent expansion in MPE V/E. For this reason, SPOOK tapes from MPE IV and MPE V/P have a different format from the SPOOK tapes of MPE V/E. The spool files themselves have remained the same in all versions.

SPOOK5, which runs only on MPE V/E systems, is able to read MPE IV, MPE V/P (E/F.00.00), and MPE V/E SPOOK tapes. Therefore, all present or past SPOOK tapes from any of your systems can be read by SPOOK5. For convenience, the System Manager can establish a UDC for SPOOK5 as described above.

If you wish to transfer spool files created on an MPE V/E system to an MPE V/P (E/F.00.00) or MPE IV system, the program SPOOK4.PUB.SYS must reside on the MPE V/P or MPE IV system. For convenience, SPOOK4 is shipped with every MPE V/E system, even though it is not run on MPE V/E. To transfer SPOOK4 to your non-MPE V/E system, store SPOOK4.PUB.SYS from your MPE V/E system onto tape, and restore it onto the MPE V/P (E/F.00.00) or MPE IV system. The System Manager can establish a "SPOOK" UDC to :RUN SPOOK4.PUB.SYS, allowing users to use the same command to run SPOOK on all systems.

SPOOK Commands

SHOW	Lists characteristics of input/output spool files.
TEXT	Accesses (open) an output spool file to permit listing its contents (>LIST command) or locating records within it (>FIND command).
LIST	Lists the contents of all or part of an output spool file made accessed by the >TEXT command.
FIND	Locates a character string in an output spool file accessed by the >TEXT command, and lists this record.
PURGE	Delete an output spool file.
APPEND	Appends part or all of an output spool file to another output spool file.
COPY	Copies part or all of an output spool file to another output spool file.
MODE	Modifies the maximum width and requests a control information display for lines listed by >LIST or >FIND commands.
ALTER	Changes the output priority, number of copies requested, and/or destination device for an output spool file.
EXIT or QUIT	Terminates operation of SPOOK.

HELP or XPLAIN	Lists all SPOOK commands and describes their syntax.
DEBUG	Enters the MPE Debug Facility. (Users with Privileged Mode (PM) capability only).
OUTPUT	Stores output spool files on tape or serial disc. (System Managers and System Supervisors only.)
INPUT	Restores output spool files from serial storage to the system. (System Managers and Supervisors only.)
RUN	Executes a prepared program.
KILL	Deletes a son process of SPOOK.

Any MPE command that can be accessed programmatically (by the COMMAND intrinsic) can be accessed by SPOOK. For a complete list of these MPE commands, refer to the MPE V Commands Reference Manual (32033-90006).

Running SPOOK

To run SPOOK5, enter the following;

```
:RUN SPOOK5.PUB.SYS
```

The program identifies itself and displays a ">" prompt:

```
SPOOK G.00.00 (C) HEWLETT-PACKARD CO., 1983.  
>
```

Entering SPOOK Commands

The following pages list syntax, parameters, operation, and examples of the SPOOK commands. There must be no space between the ">" prompt and the first letter of a command. Each command can contain up to 80 characters. Blanks may appear anywhere between syntactic elements such as names, separators, or keywords, but not within a command name. A qualified file name or user name, such as *username.acctname*, cannot contain embedded blanks.

On most terminals, press X^C to delete the current line, and Y^C to terminate the current operation.

Spool files are usually referenced by device file ID or by owner. The device file ID is a unique identifier in the form #Innn for input files, and #Onnn for output files. (The #0 is optional for output spool files.) The owner is denoted by the *username.acctname* of the creator, as reflected in the file label.

SPOOK restricts access to spool files according to the user's logon capability:

- System Managers or System Supervisors can access any spool file.
- Account Managers can access any spool file in the logon account.
- Standard Users can access any spool file that they created.

Definition of Syntax for a Range List

When a SPOOK function calls for a [,range], you may specify "ALL", or may use the following syntax to specify a subset of the file:

```
{recnumber1}          {recnumber2}  
{*                }  [+offset] [ / {*                }  [+offset] ]  
{FIRST           }  [+offset] [ / {FIRST           }  [+offset] ]  
{LAST            }                                     {LAST            }
```

where:

recnumber Is an absolute record number, or line of text in the file.
offset Specifies a record number relative to *recnumber*.

Examples of Range Lists

```
58/58+19  
22/LAST  
FIRST/LAST  
ALL  
FIRST+22/LAST  
29/82
```

>ALTER

Changes output priority, number of copies requested, or destination device for output spool file.

SYNTAX

```
>ALTER {username[.acctname]}  
      {dfid[,dfid] ... }  
  
      {PRI=outpri    }  
      [;{COPIES=copies }  
      {DEV={ldev    }}  
      {devclass}
```

PARAMETERS

<i>username</i>	Name of creator of file to alter, or "@" to alter all files of all users.
<i>acctname</i>	Account name of creator of file to alter, or "@" to alter files of all accounts.
<i>dfid</i>	Identifier of output spool file to alter, in the form # <i>0nnn</i> (#0 is optional). To reference the current spool file accessed by the >TEXT command, enter an asterisk (*).
<i>outpri</i>	The intended priority of the spool file being altered, from 1 (lowest priority) to 13 (highest priority). When PRI=1, output is always deferred. Use PRI=2 or greater if you want the spool file to be scheduled for output.
<i>copies</i>	An integer from 1 to 255, indicating the number of copies to be printed.
<i>ldev</i>	The logical device number of the device on which the spool file will be printed.
<i>devclass</i>	The device class name of the device on which the spool file should be printed.

OPERATION

>ALTER can be abbreviated >A.

PRI, COPIES, and DEV can be abbreviated P, C, and D, respectively. If you omit any PRI, COPIES, or DEV parameter, the corresponding attribute of the spool file remains unchanged.

>ALTER only operates on output spool files in the READY, OPEN, or LOCKED state, or a file currently accessed by the >TEXT command.

EXAMPLES

To change the currently scheduled number of output copies to 3 for spool file #025, enter:

```
>ALTER #025; COPIES=3
```

To change the output priority of a spool file currently accessed by the >TEXT command to 7, and the destination device to the device with the class name LP, enter:

```
>A *; P=7, D=LP
```

To change the output priority of all spool files to 8, enter:

```
>A @.@;P=8
```

To change the priority of all spool files from the PAYROLL account to 6, enter:

```
>A @.PAYROLL; P=6
```

Appends all or part of a spool file to another file.

SYNTAX

```
>APPEND {username[.acctname]}  
        {dfid[,dfid] ... } [;{range}] [,{count}] [,{filename}]  
        {END} }
```

PARAMETERS

<i>username</i>	Name of user who created the file(s) to append.
<i>acctname</i>	Account name of the user who created the file(s) to append.
<i>dfid</i>	Identifier of output spool file to append; in the form # <i>nnnn</i> .
<i>range</i>	Specific line numbers of the file to append. Refer to "DEFINITION OF SYNTAX FOR A RANGE LIST" at the beginning of this section.
<i>count</i>	The number of lines from the file to append.
<i>filename</i>	The spool file to which the file is to be appended; can be \$STDLIST, or can refer to a file equation if in the form * <i>filereference</i> . If <i>filename</i> is omitted, and no previous <i>filename</i> is currently in effect, >APPEND opens and names a new spool file.
END	Closes the new file.

OPERATION

>APPEND joins part or all of a spool file to another file. >APPEND opens a new spool file if *filename* is omitted the first time >APPEND is used. The same file remains open and subject to appending until an >APPEND END command, COPY command, or termination of SPOOK. >APPEND can be abbreviated >APPEN, APPE, APP, or >AP.

EXAMPLES

To append two spool files:

<u>>T #021</u>	Text in output file number 21.
<u>>APPEND ALL</u>	The new spool file is still open.
<u>>T #022</u>	Text in output file number 22.
<u>>AP ALL</u>	The next spool file is appended.
<u>>APPEND END</u>	The new spool file is closed.
<u>>PURGE 21,22</u>	Purge originals.

To append all files created by BRUCE.MPEUTIL, and have them put on tape:

```
>FILE TAPE; DEV=TAPE    Most MPE commands are available in SPOOK.  
>APPEND BRUCE.MPEUTIL; ALL, *TAPE  
>APPEND END
```

To append #022 to #021:

```
>APPEND #021, #022; ALL  
>APPEND END
```

Copies all or part of a spool file to another file.

SYNTAX

```
>COPY [username[.acctname]] [;[{range }  
[dfid[,dfid] ... ] [;[{count} [,filename]]
```

PARAMETERS

<i>username</i>	Name of user who created the file(s) to copy, or "*" to copy files of all users.
<i>acctname</i>	Account name of the user who created the file(s) to copy, or "*" to copy files of all accounts.
<i>dfid</i>	Identifier of output spool file to copy, in the form #0nnn.
<i>range</i>	Specific line numbers of the file to copy. Refer to Definition of Syntax for a Range List at the beginning of this section.
<i>count</i>	The number of lines of the file to copy.

OPERATION

COPY copies part or all of a spool file to another file. It closes the new file when finished. COPY can be abbreviated COP, CO, or C.

If no *username* or *dfid* is specified, this function copies the currently texted file to a new file.

EXAMPLES

To copy spool file #0123 to a permanent file called REPORT:

```
>FILE REPORT;DEV=DISC;REC=-80,,F,ASCII
>T #0123
>COPY ALL,*REPORT
```

To split one file into two, and purge the original:

```
>SHOW 293 Determine the file's length.
#FILE #JOB FNAME STATE DEV/CL PR COP RFN OWNER
#0293 #S1303 LOADMAP LOCKED LP 8 1 USER.ACCT
#FILE LDEV LABEL SECTORS LINES TIME
#0293 %5 %531461 24 77 9:50 7/7/82
```

```
>TEXT #0293 Text in the source spool file.
>COPY FIRST/36
>COPY 37/LAST
```

```
>SHOW Verify the creation of the split files.
#FILE #JOB FNAME STATE OWNER
#0293 #S1303 LOADMAP LOCKED USER.ACCT
#0295 #S1303 LOADMAP READY USER.ACCT
#0297 #S1303 LOADMAP READY USER.ACCT
```

```
>PURGE * Purge texted file.
#FILE #JOB DEV/CL SECTORS OWNER
#0293 #S1303 LP 24 USER.ACCT
```

To copy #021 and #022 to new spool files:

```
>COPY #021, #022; ALL
```

To add environment file information to spool file #0531, use >COPY to create a new spool file:

```
>FILE Q; DEV=HP2680; ENV=envfilename
>COPY #0531;ALL,*Q
```

(You may replace HP2680 with any device class name that specifies the device using the environment file, depending upon the configuration of your system.)

Invokes the MPE DEBUG facility.

SYNTAX

>DEBUG

OPERATION

DEBUG is used primarily by system programmers. DEBUG allows programmers to set breakpoints within programs, and display and modify data stacks and registers. DEBUG can set breakpoints within the SPOOK program and manipulate SPOOK's data stack. Refer to the MPE Debug/Stack Dump Reference Manual (30000-90012) for details.

DEBUG is available to you only if you have Privileged Mode (PM) capability. When accessed from SPOOK, DEBUG runs in Privileged Mode.

CAUTION

The normal safeguards of MPE are bypassed in privileged mode. When attempting to modify privileged data on disc, it is possible to destroy file integrity, or the MPE operating system itself. HP will investigate and attempt to resolve problems resulting from modification of privileged data, but this service is not included in the standard service contract. HP will not modify the MPE operating system to accommodate problems arising from your use of DEBUG.

>EXIT

Terminates operation of SPOOK.

SYNTAX

>EXIT

OPERATION

EXIT closes any open text file and terminates SPOOK. If this execution of SPOOK is a son process, SPOOK suspends and returns to the father process.

Locates a character string in a file.

SYNTAX

```
>FIND [@] ["string"] [,count]
           [,range]
```

PARAMETERS

- @** Specifies that entire lines are to be searched, rather than just leading characters.
- string** The search key to be located by >FIND. May be any number of alphanumeric characters, enclosed in quotation marks. If omitted, FIND locates the first record (line) of the file.
- count** The number of positions to search past the occurrence of *string*.
- range** The area of the file in which to conduct the search. Refer to Definition of Syntax for a Range List at the beginning of this section.

OPERATION

>FIND scans the file previously accessed by >TEXT, and locates the first record containing a string matching the *string* parameter. SPOOK can scan selected parts of the file. Leading blanks and control characters in the *string* parameter and file records are ignored. When the string is found, >FIND displays the record containing the string and the record number. If the string is not found, >FIND sets its current record pointer to one record beyond the last record scanned.

For each record located, >FIND displays:

- Line number. The record number where the string was found.
- I/O Control. I/O controls applying to the file, displayed only if requested by the MODE command. *Wnnn* indicates "write with control *nnn*"; *Cnnn* indicates "control using *nnn*"; FOPEN indicates an open file; and FCLOSE indicates a closed file.
- Text. Any leading control character is removed; nonprinting characters are each replaced by a period. If the display length has been reduced with the MODE command, the record text might appear truncated.

Line Number	I/O Control	Text (line contents)
↓	↓	↓
7	W%320	And there they met the Beast, a ponderous ...

EXAMPLES

To search spool file #0250 for the record with leading characters "NO. ERRORS":

```
>TEXT #0250  
>FIND "NO. ERRORS", ALL
```

Now, within records 21 to 30 of the same file, find the first record with ":" appearing anywhere in the line:

```
>F @ ":", 21/30
```

Now, from the current record to the end, scan leading characters for ":EOD":

```
>F ":EOD", */last
```

Lists and describes SPOOK and MPE commands.

SYNTAX

```
{MPE          }
{HELP        }
>HELP [{tablecontents  }]
      {command, [keyword]}
      {ALL          }
```

PARAMETERS

MPE	SPOOK enters the MPE help subsystem.
HELP	Displays information about the MPE help facility.
<i>tablecontents</i>	To request general MPE information, <i>tablecontents</i> can be SESSIONS, JOBS, PROGRAMS, FILES, MANAGE, or UTILITY.
<i>command</i>	Can be any MPE command.
<i>keyword</i>	Can be PARMS, OPERATION, or EXAMPLE for information about the command parameter, operation, or an example, or ALL for all three.
ALL	Displays the table of contents of the MPE help facility.

OPERATION

Do not abbreviate the word HELP.

If no parameters are specified, SPOOK describes its own commands.

The MPE HELP facility uses the same ">" prompt as the SPOOK subsystem, and the user may be unsure about which he is in. To check, enter Z (a character which is not recognized by either). If you are in SPOOK, the "INVALID COMMAND" message will appear. If you are in the MPE HELP facility, you will see:

```
Can't find anything under this command or in the table of contents.
```

To exit from the MPE HELP facility and return to SPOOK, type E, EXIT, or END.

>INPUT

Restores output spool file from serial storage to system.

SYNTAX

```
>INPUT [username[.acctname]]  
       [dfid[,dfid] ... ] ;*tapefile
```

PARAMETERS

username User name of creator of file(s) to restore, or "@" to restore the files of all users.

acctname Account name of the creator of the file(s) to restore, or "@" to restore the files of all accounts.

dfid Device file identification of the output file to restore, in the form #*0nnn*.

If *username.acctname* and *dfid* are both omitted, all files belonging to the current user are restored.

tapefile The tape file from which the input comes. A file equation must exist, and *tapefile* must be back-referenced with a "*" in the form **tapefile*.

OPERATION

>INPUT reads files back onto the system disc that were previously stored on tape or serial disc by the SPOOK >OUTPUT command. The user of >INPUT must have System Manager or System Supervisor capability.

Even if you refer to files by *dfid*, their creator names must still be defined when you execute the >INPUT command. >INPUT handles single and multi-reel tape files, and serial disc files.

After a file is input, it is assigned a new device file ID, and the job or session number of its creator is flagged with an apostrophe as follows: #S'*nnn* or #J'*nnn*

The apostrophe indicates the file was copied into the system from any external source. The device class name and logical device number of the file will change if the file is restored to a different type device from the one on which it originated.

For each spool file restored, SPOOK displays its old and new device file ID's, the new job or session number, the new logical device/class name, and the creator, in the form:

#FILE	==>	#FILE	#JOB	DEV/CL	OWNER
#099	==>	#0105	#5'44	LP	BRUCE.MPEUTIL
#0100	==>	#0106	#S'44	LP	BRUCE.MPEUTIL

EXAMPLES

To restore all files from tape file T:

```
>FILE T; DEV=TAPE  
>INPUT @.@; *T
```

To restore files #03, #04, and #07 from the same tape:

```
>I 3,4,7; *T
```

To restore all files from the same tape belonging to BILL in the JONES account:

```
>IN BILL.JONES; *T
```

>KILL

Deletes a son process.

SYNTAX

```
>KILL
```

OPERATION

>KILL deletes the son process created by the >RUN command, and all its descendants.

Lists lines of a file.

SYNTAX

```
>LIST [range ] [,count]
```

PARAMETERS

- range* The range of lines in the file to list. Refer to the "DEFINITION OF SYNTAX FOR A RANGE LIST" for more information..
- count* The number of lines to list, starting with the current line pointer.

OPERATION

For each record, >LIST displays:

- Line number. The record number where the string was found.
- I/O Control. I/O controls applying to the file, displayed only if requested by the MODE command. *w**nnn* indicates "write with control *nnn*"; *C**nnn* indicates "control using *nnn*"; FOPEN indicates an open file; and FCLOSE indicates a closed file.
- Text. Any leading control character is removed; nonprinting characters are each replaced by a period. If the display length has been reduced with the MODE command, the record text might appear truncated.

If your spool file contains HP2680 Page Printer environment file information, you must use the MODE command to enable the display of nonprinting control characters. An example of environment data appended to a spool file:

Line number	I/O Control	Control Information
0	FOPEN	
1	FDEVICECONTROL	FUNC=138 P1=% 000000 P2=% 000000 LEN=132 Download/Delete VFC
2	FDEVICECONTROL	FUNC=138 P1=% 000000 P2=% 100000 LEN= 20 Download/Delete VFC
3	FDEVICECONTROL	FUNC=135 P1=% 000000 P2=% 000000 LEN=132 Download/Delete Forms
4	FDEVICECONTROL	FUNC=135 P1=% 000000 P2=% 100000 LEN=132 Download/Delete Forms
6	FDEVICECONTROL	FUNC=135 P1=% 000000 P2=% 100000 LEN=132 Download/Delete Forms

Record text
60 W
61 W
62 W
63 W
64 W MPE IV C.00.P8
65 W 1 ININ
62 UDC (62) RJEMISC2 (162)
63 USER (63) RJESLCP (163)

When the listing is complete, the current record pointer is set to point at the record after the last listed record.

EXAMPLES

To list all records in output file #016:

```
>TEXT #016
>LIST ALL
```

If the same file contains environment file information:

```
>TEXT #016
>MODE CONTROLS=ON
>LIST ALL
```

To display the current record of this same file:

```
>L
```

To list all records from two records past the current record, up to the fourth-before-last record:

```
>L *-2/LAST-4
```

To list a total of three lines, starting with record 5:

```
>L 5, 2
```

(This actually lists lines five, six, seven, and eight.)

Adjusts line width, and enables display of control data from >FIND or >LIST commands.

SYNTAX

```
>MODE [WIDTH={+nnn}  
        {OFF }] [CONTROLS={ON }  
                {OFF}]
```

PARAMETERS

<i>nnn</i>	A signed integer indicating the width of output from >FIND or >LIST commands. A positive sign indicates width in words; negative sign indicates bytes.
OFF	WIDTH=OFF lets SPOOK use the actual source record width. CONTROLS=OFF disables display of I/O control characters.
ON	CONTROLS=ON enables display of I/O control characters resulting from >LIST and >FIND commands.

OPERATION

The initial value of WIDTH and CONTROLS is OFF. Both settings remain in effect until changed with the >MODE command.

If the spool file under observation contains HP2680 Page Printer environment information, enter MODE CONTROLS=ON before using the >LIST command.

EXAMPLES

To set the maximum line width to 60 characters (bytes), and enable display of control characters:

```
>MODE WIDTH=-60 CONTROLS=ON
```

To set the display line width to the actual record size:

```
>M W=OFF
```

Commands from the first example remain set until disabled by:

```
>M C=OFF
```


>OUTPUT

Stores output spool files on tape or serial disc.

SYNTAX

```
>OUTPUT [username[.acctname]]  
        [dfid[,dfid] ... ] ;*tapefile [,PURGE]
```

PARAMETERS

- username* User name under which the spool files to be stored were created, or "@" to store files of all users.
- acctname* Account name under which the spool files to be stored were created, or "@" to store the files of all accounts.
- dfid* Identifies the output spool file to store, in the form #*nnnn*.
- tapefile* The file to which to store the spool files. Use "*" to back-reference an existing MPE file equation.
- PURGE Indicates that files should be purged from system after being stored.

OPERATION

>OUTPUT lets the System Manager or System Supervisor copy output spool files onto tape or serial disc. These can be restored to the system with the >INPUT command. Only READY and LOCKED spool files can be output. If files are being output to multiple tape reels, the operator is prompted:

```
CHANGE REELS ON LDEV nn? YES or NO
```

To change reels or discs, the operator enters Y and mounts the new medium. To abort the operation, the operator enters N. If the operator mounts the wrong medium, SPOOK displays:

```
INCORRECT REEL - TRY AGAIN? YES or NO
```

As each file is copied to external storage, its output priority is set to 1 (deferring output). Before using >OUTPUT on that file again, use the >ALTER command to raise its output priority.

For each spool file stored, SPOOK lists the current device file ID, the session or job number of the file's creator, its logical device number or class, the number of sectors in the file, and the creator's name and account, in the form:

#FILE	#JOB	DEV/CL	SECTORS	OWNER
#099	#S44	LP	8	BRUCE.MPEUTIL
#0100	#S44	LP	12	BRUCE.MPEUTIL

Before using the >OUTPUT command, set up an MPE file equation with these parameters:

:FILE *formaldesignator* [=filereference] ;DEV=device

The ;DEV= parameter must indicate the device class name or logical device number of a magnetic tape unit. All other parameters, such as ;REC= and ;ACC=, are supplied by the >OUTPUT command; if you attempt to supply any of these, SPOOK rejects the >OUTPUT command.

EXAMPLE

To store all output spool files created by the logon user to the tape file T:

>FILE T; DEV=TAPE
>>OUTPUT @.@; *T

>PURGE

Deletes output spool files from the system.

SYNTAX

```
>PURGE {dfid} [,dfid] ...  
      { * }
```

PARAMETERS

dfid Identifies an output spool file in the form #*0nnn*.

* Purges the file currently accessed by >TEXT.

OPERATION

>PURGE can be abbreviated >P.

Deletes one or more spool files that are in the READY or LOCKED state, or the file currently accessed by >TEXT. For each spool file purged, SPOOK lists its device file ID, its creator's session or job number, its logical device/class, the number of sectors in the file, and the creator's name and account, in the form:

#FILE	#JOB	DEV/CL	SECTORS	OWNER
#0367	#S156	LP	8	BRUCE.MPEUTIL
#0368	#S156	LP	12	BRUCE.MPEUTIL
#0412	#S163	EPOC	32	GUY.MPEM

EXAMPLES

To delete spool file #024:

```
>PURGE #024
```

To delete spool files #019, #020, and #021:

```
>P 19,20,21
```

Terminates execution of SPOOK.

SYNTAX

```
>QUIT
```

OPERATION

>QUIT can be abbreviated >Q.

>QUIT closes any open text file and terminates SPOOK.

>RUN

Executes a program file

SYNTAX

```
>RUN [progfile]  
      [*      ]
```

PARAMETERS

progfile Designator of the prepared program file; can be a fully qualified file name.

* Specifies the program file previously >RUN but now suspended.

OPERATION

When you run a program with SPOOK's >RUN command, a nonzero parameter is passed to the program to be run. Therefore, do not use SPOOK to run a program that cannot accept such a parameter.

Also, do not use the ;PARM= parameter with SPOOK's >RUN command. The ;PARM= parameter is part of the MPE :RUN command, not SPOOK's >RUN command.

SPOOK can be executed from within itself. The prompt at each sublevel of SPOOK is numbered, such as:

```
>(1)  
>(2)
```

The >EXIT command suspends SPOOK, keeping its file pointer intact. Therefore, users can alternate between two active files.

>QUIT causes SPOOK to terminate and return to the father process, while >KILL causes the father process to terminate its son process.

EXAMPLES

To run a program file:

```
>RUN EDIT.TIM.DOWN
```

To run another session of SPOOK for the purpose of texting two spool files at the same time:

```
:RUN SPOOK5.PUB.SYS
```

```
SPOOK5 G.00.00 (C) HEWLETT-PACKARD CO., 1983
```

```
> T #0111 Text in first file.
```

```
.
```

```
Use it, and leave it texted.
```

```
.
```

```
>RUN SPOOK5.PUB.SYS Run "nested" program.
```

```
SPOOK5 G.00.00 (C) HEWLETT-PACKARD CO., 1983
```

```
>(1)T #0222 Text in second file.
```

```
.
```

```
Use it.
```

```
.
```

```
>(1)E Exit nested program.
```

```
SPOOK5 G.00.00 (C) HEWLETT-PACKARD CO., 1983
```

```
> E Exit SPOOK.
```

```
END OF PROGRAM
```

```
:
```

>SHOW

Lists characteristics of input or output spool files.

SYNTAX

```
>SHOW [[username[.acctname]]  
      [ dfid [,dfid] ... ]
```

```
      [@]  
      [;[I] ... ]  
      [O]
```

PARAMETERS

username The user name of the creator of the file to show, or "@" for the files of all users.

acctname The account name of the creator of the file to show, or "@" for the files of all accounts.

@ Requests the output in long format. The default format is short.

I Requests a display of input spool files only.

O Requests a display of output spool files only.

dfid Identifies the spool file to display, in the form #*Onnn* for output files, and #*Innn* for input files. Enter an asterisk (*) to refer to the currently texted file. When *dfid* is specified, output is always in long format.

OPERATION

>SHOW can be abbreviated >S.

The short format includes the device file ID, the session or job number of the creator, the file name, and the file state:

#FILE	#JOB	FNAME	STATE	OWNER
#0367	#S156	LISTER	READY	BRUCE.MPEUTIL
#0368	#S156	SLP	ACTIVE	BRUCE.MPEUTIL
#0369	#S156	BRUFILE	READY	BRUCE.MPEUTIL

The file state can be:

ACTIVE	The file is being transmitted to or from a storage input, or output device.
READY	An input file is ready for use by a program, or an output file is stored on disc and ready for output.
OPENED	The input or output spool file is being accessed by a program.
LOCKED	The output spool file is held by the system for exclusive access.

The long format appears as follows:

#FILE	#JOB	FNAME	STATE	DEV/CL	PR	COP	RFN	OWNER
#0367	#S156	LISTER	READY	LP	1	1		BRUCE.MPEUTIL
#0368	#S156	SLP	ACTIVE	PP	1	1		BRUCE.MPEUTIL
#0369	#S156	BRUFILE	READY	LP	1	1		BRUCE.MPEUTIL
#FILE	LDEV	LABEL	SECTORS	LINES	TIME			
#0367	%2	%205277	12	34	12:55		9/9/82	
#0368	%3	%130315	12	245	12:59		9/9/82	
#0369	%4	%200360	12	34	13:11		9/9/82	

The long format includes everything in the short format, plus:

- The logical device number or class name of originating device (for input files) or the destination device (for output files).
- The number of copies requested (for output files).
- RFN, where R indicates a restartable spooled job file, F indicates that a forms alignment message applies to the spool file, and N indicates that insufficient disc space was available when the spool file was created.
- The logical device number of the device where the file label is stored.
- The sector address of the spool file label.
- The total number of disc sectors used by the file.
- The total number of records (lines) in the spool file.
- The time the spool file entered the READY state.

EXAMPLES

To display, in detailed format, the characteristics of all spool files created by SMITH.ALPHA:

```
>SHOW SMITH.ALPHA; @
```

To display, in short format, the characteristics of all spool files created by the current user only:

```
>S
```

To show the characteristics of all output spool files:

```
>S @.@; 0
```

To show the characteristics of output files #023 and #026, and input file #I22:

```
>S #I12, #032,23
```


>TEXT

Accesses an output spool file for use by >FIND or >LIST.

SYNTAX

```
>TEXT dfid
```

PARAMETERS

dfid Device file ID, which identifies the output spool file, in the form *Qnnn*.

OPERATION

>TEXT accesses a READY output spool file, converts it to LOCKED state, and permits access to it with the COPY, APPEND, FIND, and LIST commands. If you text a file that is already accessed by >TEXT, the file closes and returns to READY state. When you terminate SPOOK, any texted file is automatically closed.

EXAMPLE

To open spool file #022 for access by other SPOOK commands:

```
>TEXT #022
```

or

```
>T #022
```

or

```
>T 22
```

Lists and describes SPOOK command syntax.

SYNTAX

```
>XPLAIN
```

OPERATION

>XPLAIN can be abbreviated >X.

The >XPLAIN command displays this information:

```
EXIT <<TERMINATE IF NOT A SON PROCESS>>
XPLAIN
SHOW [ USER [ .ACCOUNT ] ] [ ; [ @ ] [ ! ] [ 0 ] ]
SHOW DEVICEFILEID [ , DEVICEFILEID ]....
TEXT DEVICEFILEID
LIST [ RANGE ]
FIND [ @ ] [ "STRING" ] [ , FRANGE ]
MODE [ OPTION [ , OPTION ]...]
      OPTION = WIDTH / CONTROLS
ALTER {DFID [ ,DFID [ ,... ] ] } [ ; OPTION [ , OPTION ]....]
ALTER {USER [ .ACCOUNT ] } [ ; OPTION [ , OPTION ]....]
      OPTION = PRI / COPIES / DEV
PURGE DEVICEFILEID [ , DEVICEFILEID ]....
INPUT [ USER [ .ACCOUNT ] ] ; TAPEFILE
INPUT DEVICEFILEID [ , DEVICEFILEID ].. ; TAPEFILE
OUTPUT [ USER [ .ACCOUNT ] ] ; TAPEFILE [ ; PURGE]
OUTPUT DEVFILEID [ , DEVFILEID ] .. ; TAPEFILE [ ; PURGE]
HELP
RUN PROGRAMFILENAME [ .GROUP [ .ACCOUNT ] ]
KILL << SON PROCESS >>
QUIT << TERMINATE >>
COPY [RANGE] [,FILENAME]
COPY [DFID [ ,DFID [ ,... ] ] ;] [RANGE [,FILENAME]]
COPY [USER [ .ACCOUNT ] ;] [RANGE [,FILENAME]]
APPEND [RANGE [,FILENAME]]
APPEND [DFID [ ,DFID [ ,... ] ] ;] [RANGE [,FILENAME]]
APPEND [USER [ .ACCOUNT ] ;] [RANGE [,FILENAME]]
      [END ]
```

SPOOK2/SPOOK5 WARNING AND ERROR MESSAGES

A warning message indicates an inability to initiate a requested command, or truncation of some output. SPOOK warnings are in the following form, where *msgno* is a two digit code for the message and *message* the message text.

*WARNING = *msgno* * *message*

An error message indicates illegal syntax or parameters, and occurs immediately after command entry. Error messages also can appear after displays from >SHOW, >PURGE, >INPUT, and >OUTPUT commands. They indicate that an operation did not succeed on a certain spool file. Errors appear in the format:

[#FILE] *ERROR=*errornum* [BYTE=*byteno*] * *message*

#FILE appears only in messages errors about output files; *errornum* is the two digit error number; *byteno* identifies the character causing the error; and *message* is the text of the message.

Warning Messages

- | | | |
|---|----------------------------|--|
| 1 | NOT INTERACTIVE
SESSION | SPOOK must be run interactively, not as a batch job. |
| 2 | END OF FILE | SPOOK encountered an end-of-file mark in your response, such as a colon (:) in column one. Run SPOOK again and respond correctly. |
| 3 | TOO MANY FILES | You have asked SPOOK to handle too many files. in one operation. Reissue the command for fewer files. |
| 4 | INSUFFICIENT
CAPABILITY | You lack an MPE user capability to perform a command, such as System Manager (SM) or Privileged Mode (PM). Call your System Manager, or try another command. |

Error Messages

- | | | |
|----|------------------------------|---|
| 19 | IMPOSSIBLE
INTERNAL ERROR | SPOOK system error. Call your HP System Engineer. |
| 20 | INVALID COMMAND
NAME | Re-enter the command with correct spelling and syntax. |
| 21 | COMMAND NAME
TOO BIG | Re-enter the command with correct spelling and syntax. |
| 22 | PROMPT I/O
ERROR | An input/output system transmission error occurred while SPOOK was sending its prompt to your terminal. SPOOK will try again. |

23	INPUT I/O ERROR	An input/output system transmission error occurred while SPOOK was reading from the terminal. SPOOK will try again.
24	UNABLE TO CLOSE FILE	SPOOK cannot close a spool file, e.g. it cannot return a currently texted file to the system after the execution of a new >TEXT command. Re-enter command.
25	UNABLE TO PURGE FILE	SPOOK cannot delete the spool file referenced in the >PURGE command. Type the >PURGE command again.
26	FILE READ ERROR	SPOOK cannot access the texted file. Use >TEXT again, then retry your >FIND or >LIST command.
27	FILE FCONTROL ERROR	SPOOK cannot perform an input/output control operation, such as rewinding the tape. Retry your command.
28	FILE NOT 'READY'	You attempted to access a spool file not in the READY state. Wait until the state changes.
29	UNABLE TO OPEN FILE	To >TEXT or >PURGE a file, it must be in the READY state.
30	INPUT FILE NOT ALLOWED	You cannot access an input spool file in this context.
31	FILE NOT FOUND	The requested file does not exist. Check the device file ID and spelling.
32	INVALID FILE ID	The requested device file ID does not exist. Check your syntax, or use >SHOW to verify the ID.
33	UNEXPECTED CHARACTER	Input syntax error. Try again.
34	USER NAME TOO BIG	User names must be eight characters or less. Check name and retry.
35	USER NOT ACCESSIBLE	The requested user name does not exist, or cannot be accessed by you. Check spelling.
36	ACCOUNT NAME TOO BIG	Account names must be eight characters or less. Check spelling.
37	ACCOUNT NOT ACCESSIBLE	The requested account name does not exist, or is inaccessible by you.
38	INVALID LINE MNEMONIC	Check your syntax, and the spelling of FIRST and LAST in the range specification.
39	INVALID LINE NUMBER	The specified line number does not exist in file. Check syntax.
40	INVALID LINE COUNT	The count parameter is incorrect. Check syntax.
41	INVALID LINE RANGE	Your range specification does not reference any lines in the file.
42	NON TERMINATED CHARACTER STRING	At least one quotation mark is missing. Check syntax.

43	INVALID OPTION NAME	Illegal keyword in the >MODE or >ALTER command. Check syntax and spelling.
44	INVALID OPTION SEPARATOR	The equal sign (=) is missing. Check syntax.
45	INVALID OPTION PARAMETER	An illegal parameter follows the keyword. Check syntax and spelling.
46	NO TEXT FILE	You must >TEXT a file before using >FIND or >LIST.
47	FILE NOT 'READY/OPEN'	>ALTER must work on READY or OPEN files. You cannot alter the file because it is LOCKED or ACTIVE.
48	TEXT FILE NOT ALLOWED	You may not access the currently texted file with this command.
49	MISSING SEMICOLON	Check the syntax.
50	UNABLE TO OPEN TAPE FILE	Be sure the tape is properly mounted and the tape unit is on line.
51	UNABLE TO CLOSE TAPE FILE	Dismount the tape manually. This automatically closes the tape file.
52	INVALID TAPE FILE	The file designator is bad. Check your file equations and be sure file was correctly back-referenced with a "*".
53	INVALID TAPE FORMAT	You cannot >INPUT a file that was not previously >OUTPUT to this tape. Be sure the tape is correct.
54	TAPE FILE READ ERROR	Try again.
55	TAPE FILE WRITE ERROR	Try again.
56	USER.ACCOUNT NOT ALLOWED	Do not reference files by creator names in this command. Use a device file ID.
57	NO EQUIVALENT DEVICE	This system does not have the device SPOOK is looking for, probably because the >OUTPUT command was done on another system.
58	NO EQUIVALENT CLASS	This system does not have the device class SPOOK is looking for, probably because the >OUTPUT command was done on another system.
59	NO ROOM IN DEVICE TABLE	SPOOK cannot restore all the requested files at this time. Try again later.

- 60 MULTI REEL ABORT You aborted a multi-reel >INPUT command by responding N to:
CHANGE REELS ON LDEV *nn*? N

If this action was a mistake, re-enter the >INPUT or >OUTPUT command.
- 69 INVALID LENGTH OF RECORD IN TEXT FILE SPOOK cannot list the texted file. >TEXT again and retry. If it fails again, recreate the spool file and purge the bad one.
- 70 FILE IS NOT PROGRAM FILE The file does not exist, or is not a program file. To verify, type LISTF *filename*,2.
- 71 NO SON PROCESS TO BE DELETED Any son processes were already killed.
- 72 MISSING PROGRAM FILE NAME Check syntax.
- 73 UNABLE TO CLOSE COPY FILE (*nn*) Error (*nn*) was returned by FCLOSE. Try again.
- 74 UNABLE TO OPEN COPY FILE (*nn*) Error (*nn*) was returned by FOPEN. Refer to the description of the FCHECK intrinsic in the MPE V Ininsics Reference Manual (32033-90007) for a list of error numbers.
- 75 SPOOLFILE CREATE ERROR An attempt to open a new spool file failed.
- 76 UNABLE TO RENAME COPY FILE The copy file has same name as a permanent file.
- 77 DS COPY NOT YET This feature is not available.
- 78 LINE NUMBER IS IN PURGED EXTENT You cannot >LIST, >APPEND, or >COPY a line number in a purged extent.
- 79 INVALID COPY FILE Illegal file name. Re-enter the correct file name or reference.
- 80 MISSING DFID OR USER.ACCOUNT The device file ID or *username.accountname* must be supplied. Try again.

FREE5 details the contiguous free space on each mounted disc volume, the total free space on each disc volume, and the total free space in the system. This is an aid in determining disc usage and the degree of disc space fragmentation. All values are decimal. No special capabilities are required to run FREE5.

The FREE5 utility (FREE5.PUB.SYS) runs on MPE V/E; the MPE V/P and MPE IV version is FREE2 (FREE2.PUB.SYS). The dialog and operation of both are identical. Throughout this section, FREE5 will be used in all examples and explanations.

OPERATION

The output file for FREE5 is \$STDLIST (the terminal). To run FREE5 with output directed to the terminal, type:

```
:RUN FREE5.PUB.SYS
```

The FREE5 output can be redirected by using the formal file designator FREE5OUT. To redirect the output to a line printer, use a file equation as follows:

```
:FILE FREE5OUT;DEV=LP  
:RUN FREE5.PUB.SYS
```

To determine the amount of free space on a Private Volume, the Private Volume Set must be logically mounted using the :MOUNT command:

```
:MOUNT vcname.group.account  
:RUN FREE5.PUB.SYS
```

A logical mount prevents a physical dismount of the Private Volume. Refer to the MPE V Commands Reference Manual (32033-90006) for a description of the :MOUNT command.

EXAMPLE

To run FREE5, type:

```
:RUN FREE5.PUB.SYS
```

```
FREE5 G.00.00 (C) HEWLETT-PACKARD CO.,1983
VOLUME C11D1U0      LDEV 1 ← Volume name and logical device number.
LARGEST FREE AREA= 39848 ← Number of sectors in largest free area on this disc.
  SIZE COUNT  SPACE  AVERAGE
>100000 0     0      0
>10000  2     50104  25052
>1000   12    25062  2088
>100    28    8695   310 ← There are 28 free areas between 101 and 1000
>10     122   4020   32    sectors, with an average size of 310 sectors,
>1      181   675    3     for a total of 8695 sectors.
TOTAL FREE SPACE=88556
*****
VOLUME C11D1U1      LDEV 2
LARGEST FREE AREA= 10123
  SIZE COUNT  SPACE  AVERAGE
>100000 0     0      0
>10000  1     10123  10123
>1000   12    23597  1966
>100    26    8363   321
>10     134   3239   24
>1      197   678    3     Total of SPACE column, which shows the
TOTAL FREE SPACE=46000 ← total free space on the volume.
*****
SYSTEM TOTAL FREE SPACE=134556 ← Total free space on all volumes.

END OF PROGRAM
:
```

FREE 2/FREE5 ERROR MESSAGES

LDEV #nn HAS #mm PAGES OF
DISC FREE SPACE MAP MARKED
AS BAD. UP TO #pppp SECTORS
OF DISC SPACE MAY BE LOST.

The Disc Free Space Map (DFS_M) is damaged. You must
COOLSTART to recover lost disc space.

ALLOCATION HAS BEEN
DISABLED ON LDEV #nn.

Disc Free Space Map (DFS_M) is damaged. Existing files can be
accessed, but new files cannot be created on this device.

LDEV #nn HAS BAD DFS_M

You must COOLSTART to recover lost disc space.

LDEV ##nn NOT MOUNTED,
OR HAS NO DFS_M.

If the logical device is mounted, the disc volume is in a
pre-C0.00.08 free space format. Store the files on the disc,
convert the disc to the new format using the VINIT subsystem's
>FORMAT and >INIT commands, and restore the files.

The Memory Logging Analyzer (MEMLOGAN), lets the System Manager print the error logs that were recorded by the Memory Error Logging System.

MPE initiates MEMLOGP, the memory error logging process, when the system is initialized. Once an hour, MEMLOGP obtains error data from the memory error logging boards and writes the data to the two-record file MEMLOG.PUB.SYS. The time period may be adjusted with the MEMTIMER utility program, also described in this manual.

The Memory Logging Facility is not related to the System Logging Facility that records events on the System Log File.

OPERATION

1. To run MEMLOGAN, type:

```
:RUN MEMLOGAN.PUB.SYS [ ;PARM=n ]
```

If *n* equals:

- 0 The memory log file is printed but not cleared. This is the default value of *n*.
 - 1 This option displays and clears the memory log file. Execute MEMLOGAN with this option immediately after bringing up the system for the first time, after changing the size of memory.
 - 2 Displays the memory log file and deletes the file after the next =SHUTDOWN command. If you have error-correcting memory, a new MEMLOG will be built when the system is restarted. Errors continue to be logged in MEMLOG between the execution of this option and the actual shutdown time.
2. MEMLOGAN identifies itself and begins to output one of the error log summaries:

```
MEMLOGAN G.00.00 (C) HEWLETT-PACKARD CO., 1980  
LOGGING STARTED - DATE: 11/ 7/83 TIME: 15:16  
FIRST ERROR LOGGED - DATE: 1/ 6/84 TIME: 9:18  
LAST ERROR LOGGED - DATE: 1/ 6/84 TIME: 6:49
```

or

```
MEMLOGAN G.00.00 (C) HEWLETT-PACKARD CO., 1980  
LOGGING STARTED - DATE: 6/ 6/83 TIME: 9:00  
FIRST ERROR LOGGED - DATE: 9/15/83 TIME: 12:55  
LAST ERROR LOGGED - DATE: 3/28/84 TIME: 16:47
```

MEMLOGAN

3. When MEMLOGAN has output all of the error log, it terminates with the message:

```
END OF PROGRAM  
:
```

4. If MEMLOGAN completes with MEMLOG clear and in the "no-error" state, as after being run with ";PARAM=1", MEMLOGAN displays the following message before it terminates:

```
* NO ENTRIES IN MEMLOG FILE *
```

MEMLOGAN Environment

The default output device for MEMLOGAN is the terminal, but you can redirect the output to another device by using the file equation that references the formal file designator OUT:

```
:FILE OUT; DEV=LP  
:RUN MEMLOGAN.PUB.SYS
```

Memory Error Log Formats

The descriptions which follow show the data formats recorded by the Memory Error Logging System. There are two versions: one for the Series 39/40/42/44/48, and one for the Series 64/68. The kinds of errors recorded are the same for both series of computers. The main difference in the formats is the presence of the controller field in the data for the Series 39/40/42/44/48.

MEMLOGAN on Series 39/40/42/44/48

I	ADDRESS			I	ERROR TYPE			I	ERROR	I		
I	CONTROLLER	I	BOARD	I	ROW	I	TYPE	BIT	CHIP	I	COUNT	I
I	controller	I	board	I	row	I	type	bit	chip	I	count	I
I		I		I		I				I		I
I		I		I		I				I		I

Field	Content
<i>controller</i>	The memory controller where the error occurred.
<i>board</i>	The memory module board on which the error occurred.
<i>row</i>	The row designation on the board in which the failing chip is located.
<i>type</i>	Type of error detected, as follows: <ul style="list-style-type: none"> CHECK Check bit error. DATA Data bit error. MULTIPLE BIT ERROR Error is more than one bit. FORCED D.E.W. Forced Double Error Write. Parity error on the data transmitted to memory. MISSING ARRAY BOARD Non-responding array board.
<i>bit</i>	If <i>type</i> is CHECK, then <i>bit</i> is the failing check bit. If <i>type</i> is DATA, then <i>bit</i> refers to the failing data bit.
<i>chip</i>	The chip on which the error occurred, in the format U_n , where n is a digit indicating the chip number.
<i>count</i>	The number of logging intervals during which this error was detected at least once. This value does not represent the number of times that an error was actually detected.

MEMLOGAN on Series 64 and 68

I	ADDRESS	I	ERROR TYPE	I	ERROR	I				
I	BOARD	I	WORD	I	TYPE	BIT	CHIP	I	COUNT	I
I	<i>board</i>	I	<i>word</i>	I	<i>type</i>	<i>bit</i>	<i>chip</i>	I	<i>count</i>	I
I		I		I				I		I

Field	Content
<i>board</i>	The memory module board on which the error occurred.
<i>word</i>	The word, within the data block, where the error occurred.
<i>type</i>	Type of error detected, as follows: <ul style="list-style-type: none"> CHECK Check bit error. DATA Data bit error. MULTIPLE BIT ERROR Error is more than one bit. FORCED D.E.W. Forced Double Error Write. Parity error on the data transmitted to memory. MISSING ARRAY BOARD Non-responding array board.
<i>bit</i>	If <i>type</i> is CHECK, then <i>bit</i> is the failing check bit. If <i>type</i> is DATA, then <i>bit</i> is the failing data bit.
<i>chip</i>	The chip on which the error occurred, in the format U_n , where n is a digit indicating the chip number.
<i>count</i>	The number of logging intervals during which this error was detected at least once. This value does not represent the number of times that an error was actually detected.

EXAMPLES

If MEMLOG was updated with no errors found, MEMLOGAN prints the date and time of the first and last updates, and the logging interval at the time of the last update. The interval appears in the form of *hh:mm:ss*, indicating hours, minutes, and seconds. MEMLOGAN then terminates after displaying:

```
*** NO ERRORS LOGGED ***
```

If errors were logged, MEMLOGAN prints the date and time of the first and last logging, the date and time of the first and last logged errors, and an error table.

Series 39,40,42,44,48 MEMLOGAN Output

```
:RUN MEMLOGAN.PUB.SYS
```

```
MEMLOGAN G.00.00 (C) HEWLETT-PACKARD CO., 1980
```

```
LOGGING STARTED      - DATE:  5/ 8/83   TIME: 15:16
FIRST ERROR LOGGED   - DATE:  9/ 5/83   TIME:  9:18
LAST ERROR LOGGED    - DATE:  8/29/83  TIME:  6:49
LAST LOG UPDATE      - DATE: 10/12/83  TIME: 12:19
TIMING INTERVAL      -   1:00:00
```

```
-----
```

I	ADDRESS	I	ERROR TYPE	I	ERROR I							
I	CONTROLLER	I	BOARD	I	ROW	I	TYPE	BIT	CHIP	I	COUNT	I
I	CONTROLLER A	I	0	I	1	I	CHECK	0	U198	I	2	I
I		I		I	6	I	DATA	9	U103	I	13	I
I		I	1	I	0	I	DATA	0	U19	I	4	I
I		I		I	0	I	MULTIPLE BIT ERROR			I	1	I
I		I		I	0	I	DATA	11	u149	I	3	I
I		I	2	I	7	I	DATA	14	U172	I	2	I
I		I	3	I	3	I	CHECK	5	U246	I	3	I

```
-----
```

```
END OF PROGRAM
:
```

Series 64 and 68 MEMLOGAN Output

:RUN MEMLOGAN.PUB.SYS

MEMLOGAN G.00.00 (C) HEWLETT-PACKARD CO., 1980

LOGGING STARTED - DATE: 6/ 6/83 TIME: 9:00
 FIRST ERROR LOGGED - DATE: 9/15/83 TIME: 12:55
 LAST ERROR LOGGED - DATE: 3/28/84 TIME: 16:47
 LAST LOG UPDATE - DATE: 4/30/84 TIME: 14:05
 TIMING INTERVAL - 1:00:00

```

-----
I  ADDRESS  I      ERROR TYPE      I ERROR I
-----
I BOARD I WORD I  TYPE  BIT  CHIP  I COUNT I
-----
I  0    I  0    I DATA  10  U1202 I   1 I
I      I  2    I DATA   6  U1606 I   1 I
I  1    I  1    I DATA   2  U2004 I   1 I
I      I  2    I CHECK  2   U 807 I   1 I
I      I  3    I DATA   2  U2008 I   9 I
I  2    I  1    I DATA  25  U1705 I   1 I
I  4    I  0    I CHECK  5   U 503 I   1 I
I      I  1    I DATA  24  U1805 I   1 I
I      I  3    I DATA  11  U1108 I   1 I
I      I      I DATA  16  U 608 I   1 I
-----

```

END OF PROGRAM

:

MEMLOGAN ERROR MESSAGES

MEMLOGAN displays a message and terminates on encountering an error. If an error is detected by the MPE file system, it is explained by the File Information Display (described in the MPE V Intrinsic Reference Manual (32033-90007)), followed by one of the messages:

- * OUT FILE ERROR * For output errors.
- * LOG FILE ERROR * For errors in reading the MEMLOG.

If the error was detected by MEMLOGAN, not the operating system, MEMLOGAN displays:

- * MEMLOGAN ERROR: *errnum* *

If *errnum* is 1, MEMLOGAN could not lock MEMLOG for exclusive use. If *errnum* is 2, MEMLOGAN could not unlock the MEMLOG for access by other processes.

MEMTIMER sets the time interval between updates of the memory log file MEMLOG.PUB.SYS by the system process MEMLOGP. The user must have System Manager (SM) capability to run MEMTIMER.

The default logging interval of one hour is usually sufficient to provide an overview of system memory performance. MEMTIMER lets you request more frequent logging of memory data. The default interval is re-established every time the system is initialized. Therefore, you must run MEMTIMER after every system cold load if you wish to have a different interval.

OPERATION

1. To run MEMTIMER, type:

```
:RUN MEMTIMER.PUB.SYS; PARM=n
```

The *n* denotes the new logging interval, in seconds, from 1 to 65536.

2. MEMTIMER identifies itself and reminds you of the capability that you must have to continue. If you have the correct capability, MEMTIMER causes logging to begin:

```
MEMTIMER G.00.00 (C) HEWLETT-PACKARD CO., 1976  
** PROGRAM REQUIRES SYS. MGR. CAPABILITY **
```

3. When the logging is complete, MEMTIMER resets the interval for periodic logging and suspends itself with the message:

```
END OF PROGRAM  
:
```

EXAMPLE

To change the logging interval to ten seconds, enter:

```
:RUN MEMTIMER.PUB.SYS; PARM=10  
** PROGRAM REQUIRES SYS. MGR. CAPABILITY **  
END OF PROGRAM  
:
```


MEMTIMER

MEMTIMER ERROR MESSAGES

****INVALID PARM
(DELAY) VALUE**

You entered an invalid value for *n*. The current interval remains in effect, and MEMTIMER terminates. Try again with the correct value.

****MEMORY LOGGING
PROCESS NOT
ACTIVE****

The logging program MEMLOGP was not active. (Perhaps a file system error caused MEMLOGP to abort.) Try MEMTIMER again.

****MEMLOGP TIMER
ENTRY NOT
FOUND****

Memory logging hardware is absent from system, or logging is currently in progress. If the hardware is installed, run MEMTIMER again to be sure MEMLOGP recognizes the new interval.

LISTLOG2/LISTLOG5

SECTION

VIII

The program LISTLOG5 (LISTLOG5.PUB.SYS) runs on MPE V/E systems; the MPE V/P and MPE IV equivalent is LISTLOG2 (LISTLOG2.PUB.SYS). The dialogue and operation of the two are identical. In this section, LISTLOG5 will be used for all examples and discussion.

LISTLOG5 analyzes files on the MPE system log file. An MPE log file records events such as session or job initiation and termination, process termination, file closure, and system shutdown. Refer to the MPE V System Operation and Resource Management Reference Manual (32033-90005) for more information on system logging.

Log files are named by the following convention, where *nnnn* is a four-digit number:

LOG*nnnn*.PUB.SYS

The formal file designator of the output file is LOGLIST, with the default device class LP. LOGLIST is opened as new, and closed as a permanent file.

OPERATION

1. To find out which log files are on the system before you run LISTLOG5, enter the following:

:LISTF LOG@.PUB.SYS

MPE returns a list of numbers for all of the log files currently on the system. These are the valid numbers you can choose from when you run LISTLOG5. For example:

FILENAME

LOG	LOG1958	LOG1959	LOG1960	LOG1961	LOG1962
LOG1963	LOG1964	LOG1965	LOG1966	LOG1967	LOG1968
LOG1969	LOG1970	LOG1971	LOG1972	LOG1973	LOG1974
LOG1975	LOG1976	LOG1977	LOG1978	LOG1979	LOG1980

2. To run LISTLOG5, type:

:RUN LISTLOG5.PUB.SYS

3. LISTLOG5 identifies itself and asks for the number of the first log file to print:

LISTLOG5 G.00.00 (C) HEWLETT-PACKARD CO., 1982

ENTER FIRST AND LAST LOG FILES TO BE ANALYZED
FIRST? 1958

Enter the four-digit numbers from the list of log files. If you only want to analyze one file, enter it as the first file number and press (RETURN) in response to the "LAST?" prompt.

LISTLOG2/LISTLOG5

4. You are then prompted for the four-digit number of the last log file to print. Press **RETURN** to list only the first file.

LIST? 1980

5. LISTLOG5 now displays a numbered list of events for which histories can be printed :

TYPE NO.	EVENT
0	LOG FAILURE
1	SYSTEM UP
2	JOB INITIATION
3	JOB TERMINATION
4	PROCESS TERMINATION
5	FILE CLOSE
6	SYSTEM SHUTDOWN
7	POWER FAILURE
8	SPOOLING LOG RECORD
9	LINE DISCONNECTION
10	LINE CLOSE
11	I/O ERRORS
12	PRIVATE VOLUMES
13	PRIVATE VOLUMES
14	TAPE LABELS
15	CONSOLE LOG RECORD
16	PROGRAM FILE EVENT
17	CALL PROGRESS SIGNALS
18	DCE PROVIDED INFO
46	MAINTENANCE REQUEST
47	DIAGNOSTIC CONTROL UNIT

6. At the end of the list of events, you are prompted for input with the message:

ENTER EVENT NUMBERS SEPARATED BY COMMAS. A CARRIAGE
RETURN ASSUMES ALL EVENTS WILL BE EVALUATED

Type the event numbers and press **RETURN**. LISTLOG5 creates spool files of the events that you requested. There are no messages echoed back to your terminal if your request is successful. If your request is not successful, one of two messages will be displayed: 1) an error message in the format described under "ERROR CONDITIONS", or 2) a message indicating that there are no events for the log file numbers that you requested:

NO DESIRED EVENTS FOUND IN LOGFILE 2008

If events have been found for a log file, it's number will not appear in the "NO DESIRED EVENTS" list:

NO DESIRED EVENTS FOUND IN LOGFILE 1960
NO DESIRED EVENTS FOUND IN LOGFILE 1962

Events have been found for all other requested log files, so they do not appear in this list.

7. LISTLOG5 then asks:

DO YOU WANT TO PURGE LOG FILES?

If you answer YES, the log files are printed and then purged from the system. If you answer NO, the files are printed and also retained by the system. You are now asked if you want to rerun the program; Type YES to continue with LISTLOG5, NO or N to terminate:

DO YOU WISH TO RUN AGAIN (Y OR N)? N

EXAMPLES

To print a MPE system log file, type:

:RUN LISTLOG5.PUB.SYS

```
LISTLOG5 E.00.00 (C) HEWLETT-PACKARD CO., 1982
ENTER FIRST AND LAST LOG FILE TO BE ANALYZED
FIRST? 2824          Enter a four digit number.
LAST? 2825          Press (RETURN) if you only want to print the first file.
ENTER EVENTS TO BE PRINTED
TYPE NO.          EVENT
 0          LOG FAILURE
 1          SYSTEM UP
 2          JOB INITIATION
 3          JOB TERMINATION
 4          PROCESS TERMINATION
 5          FILE CLOSE
 6          SYSTEM SHUTDOWN
 7          POWER FAILURE
 8          SPOOLING LOG RECORD
 9          LINE DISCONNECTION
10          LINE CLOSE
11          I/O ERRORS
12          PRIVATE VOLUMES
13          PRIVATE VOLUMES
14          TAPE LABELS
15          CONSOLE LOG RECORD
16          PROGRAM FILE EVENT
17          CALL PROGRESS SIGNALS
18          DCE PROVIDED INFO
46          MAINTENANCE REQUEST
47          DIAGNOSTIC CONTROL UNIT          Series 64 and 68, only.
```

ENTER EVENT NUMBERS SEPARATED BY COMMAS. A CARRIAGE RETURN ASSUMES ALL EVENTS WILL BE EVALUATED.

11
DO YOU WANT TO PURGE LOG FILES? YES

DO YOU WISH TO RUN AGAIN (Y OR N)? N

LISTLOG2/LISTLOG5

To redirect the LISTLOG5 output to disc, then copy the contents to a line printer (LP):

```
:FILE LOGLIST=LOGFILE;DEV=DISC;REC=-132,1,F,ASCII;CCTL  
:RUN LISTLOG5.PUB.SYS  
.  
.  
.  
END OF PROGRAM  
:FILE LP;DEV=LP;CCTL  
:FCOPY FROM=LOGFILE;TO=*LP  
.  
.  
.
```

To redirect the LISTLOG5 output to tape, then copy it to a line printer (LP):

```
:FILE LOGLIST=LOGTAPE;DEV=TAPE;REC-132,1,F,ASCII;&  
:CCTL;ACC=APPEND Lets old output be stored on tape.  
:RUN LISTLOG5.PUB.SYS  
.  
.  
.  
END OF PROGRAM  
  
:FILE LP;DEV=LP;CCTL  
:FILE LOGTAPE;DEV=TAPE;REC=-132,1,F,ASCII;CCTL  
:FCOPY FROM=*LOGTAPE;TO=*LP  
.  
.  
.
```

LISTLOG2/LISTLOG5 ERROR CONDITIONS

On fatal errors, LISTLOG5 displays a message such as:

```
FOPEN ERROR TO LOGFILE xxxx ERROR NUMBER yy
```

where *xxxx* is the log file number, and *yy* is the FCHECK error code as described in the MPE V Intrinsic Reference Manual (32033-90007).

After the FOPEN error message, LISTLOG5 prompts with:

```
DO YOU WANT TO RUN AGAIN (Y OR N)?
```

If you answer YES, LISTLOG5 starts over again with:

```
ENTER FIRST AND LAST LOG FILE TO BE ANALYZED.
```

ASOCTABL/ASOCTBL5

SECTION

IX

The ASOCTBL5 utility (ASOCTBL5.PUB.SYS) runs on MPE V/E systems; the MPE V/P and MPE IV version is ASOCTABL (ASOCTABL.PUB.SYS). The operation and dialogue of both versions is identical. Throughout this section, ASOCTBL5 will be used in all examples and explanations.

ASOCTBL5 is used by the System Manager to create the device class/user association table in ASSOCIATE.PUB.SYS. To use ASOCTBL5 you must have SAVE and WRITE access to PUB.SYS, or you must be logged on to the SYS account. This table defines which users are authorized to associate with which device classes. If you are authorized in the association table, you may use the :ASSOCIATE command to gain access to a device class. By implication, this means that you can :ASSOCIATE a device class such as LP to yourself, for exclusive use. Once gained, this association lasts until you log off or :DISASSOCIATE.

In order for you to :ASSOCIATE a device, no devices in this class can be previously associated by another user. After you :ASSOCIATE a device class, you may execute any of the operator commands appropriate to this device. Messages for your associated device class appear on your \$STDLIST device.

Only one user may associate to a given device class at one time. If your device belongs to several device classes and one of these device classes has been associated to another user, you cannot use the :ASSOCIATE command. The device is unavailable and you must be able to associate all devices in all of the classes that you are using.

Input to ASOCTBL5 is from a terminal or from a file. If ASOCTBL5 reads input from a file, the formal file designator is INPUT. ASOCTBL5 reads that file until an end-of-file is encountered, or until a record starting with EXIT or exit in column 1 is found. If no file equation exists, ASOCTBL5 prompts for an input from the terminal. While scanning the input, ASOCTBL5 builds a temporary file. When it has processed all of the input, it deletes the existing ASSOCIATE file and saves the temporary file as ASSOCIATE.PUB.SYS. If errors are encountered in the input, ASOCTBL5 continues to scan the input following the error, but it doesn't delete the previous file or save the temporary file.

OPERATION

1. To run ASOCTBL5, type:

```
:RUN ASOCTBL5.PUB.SYS
```

2. ASOCTBL5 identifies itself and prompts for an input.

```
ASOCTBL5 G.00.00 (C) HEWLETT-PACKARD CO., 1979  
>
```

3. Enter a device class name, followed by "=", and a list of user names and account names in the following form:

```
> devclass = username.acctname [, ... ]
```

ASOCTABL/ASOCTBL5

For example:

```
>LP=DANL,BOONE
>TAPE=JOHN.BGOOD
>EXIT
```

The *devclass* must be a device class that exists in the current SYSDUMP configuration. When a device class is specified, all devices of that class are affected by the :ASSOCIATE command.

The *username* or *acctname* can be replaced by "@" to indicate all possible responses for that item:

```
user.@      Enables all users with the specified name in any account.
@.acctname  Enables all users in the specified account.
@.@         Enables all users.
```

For example, to allow all users in the FINANCE account, and TONY.ABC, to associate all devices of class LP:

```
>LP=@.FINANCE,TONY.ABC
```

EXAMPLES

To list the users who are authorized to associate, and the device classes they may associate with, run the program in this way:

```
:RUN ASOCTBL5,LIST
```

The file ASSOCIATE is not an Editor file, and cannot be modified directly. You may, however, use an Editor file as the input for ASOCTBL5, by using a file equation for the formal designator INPUT:

```
:EDITOR                Create the Editor file.
/ADD
  1  LP=BBALL.JONES
  2  TAPE=FIELD.SUPPORT
  3  EXIT
  4  //
/K ASOT, UNN          Keep the file unnumbered.
/E
END OF SUBSYSTEM

:FILE INPUT=ASOT      Establish the Editor file as the input file.
:RUN ASOCTBL5.PUB.SYS

LP=BBALL.JONES
TAPE=FIELD.SUPPORT

END OF PROGRAM
:
```

If you choose not to use a file equation for INPUT, ASOCTBL5 prompts for input at the terminal:

```
:RUN ASOCTBL5.PUB.SYS  
>LP=JIM.USERS  
>EXIT  
  
END OF PROGRAM  
:
```

ASOCTABL/ASOCTBL5 ERROR MESSAGES

All of these error messages denote fatal conditions. You can continue to make inputs after the appearance of one of these messages, but no modifications will be made to the association table. You need to exit from the program and retry from the beginning.

```
UNABLE TO DELETE OLD 'ASOCIATE.PUB.SYS' FILE  
  
EXPECTED AT LEAST 3 PARAMETERS, LDEV = USER.ACCT  
= MUST FOLLOW LDEV  
  
UNABLE TO OPEN INPUTFILE  
  
CLASS NAMES ARE LIMITED TO 8 CHARACTERS  
  
NO SUCH CLASS IN THIS SYSTEM  
  
EXPECTED FOLLOWING USER NAME.  
  
UNABLE TO OPEN NEW ASOCIATE.PUB.SYS FILE.
```


The DPAN5 program produces a formatted listing of main memory, based on a memory dump taken after a system failure, HALT, or other abnormal condition. The listing can be studied by your Hewlett-Packard System Engineer to obtain information about the failure. No special MPE capabilities are required to run DPAN5.

The DPAN5 utility (DPAN5.PUB.SYS) runs on MPE V/E; the MPE V/P and MPE IV version is DPAN4 (DPAN4.PUB.SYS). The dialogue and operation of both are identical. Throughout this section DPAN5 will be used in all examples and explanations.

If your system has more than 2 megabytes of memory, configure your spool files with the maximum number of sectors per extent (32767) so that they are big enough to hold all required data. In order to access the maximum extent size, you may need to use the >COND command of the VINIT subsystem, as described in the MPE V System Operation and Resource Management Reference Manual (32033-90005).

NOTE

Always save the original tape, cartridge tape, or serial disc generated in taking the dump. Your Hewlett-Packard System Engineer may need this data, along with the DPAN5 printout.

OBTAINING A DPAN5 LISTING

Follow these steps whenever the system crashes for an unknown reason:

1. Immediately after system termination, use the Software Dump Facility (SDF) to make a main-memory dump. This dump is copied onto magnetic or cartridge tape or serial disc by firmware microcode. It provides an instant "snapshot" picture of memory at the time of the termination. The SDF is described in the MPE V System Operation and Resource Management Reference Manual (32033-90005).
2. When the system has been restarted, enter the command:

:RUN DPAN5.PUB.SYS

This formats the dump based on input from the serial storage device.

Memory Dump on Series 39/40/42/44/48

CAUTION

You must have at least one backup copy of the Software Dump Facility on serial disc, magnetic tape, or cartridge tape, regardless of your configuration. A **BACKUP COPY OF SDF CANNOT BE CREATED AFTER A SOFTWARE DUMP HAS FAILED**. Create the backup when the system is initially configured, and whenever you receive a software update. Refer to the MPE V System Operation and Resource Management Reference Manual (32033-90005) for instructions on creating a backup copy of SDF.

1. Mount a serial disc, cartridge tape, or magnetic tape on a logical device specified by the device class DDUMP, then place the device on line.
2. On the System Control Panel set the DUMP thumbwheel switch to the octal value of the DRT number (channel address and device address) of the system disc drive.
3. Press the DUMP key on the System Control Panel, or enable the Control and Maintenance Processor (CMP), by pressing B. When you see the CMP prompt character (->), enter DUMP on the System Console.
4. The Software Dump Facility loads itself from the specified device and begins a serial execution of the SDF command file SDFCOM. The command file is located on the system disc and contains ASCII commands for the following:
 - Change the SDF console's DRT number.
 - Dump main memory.
 - Change the channel or device address of the dump device.
 - Obtain any remaining commands from the console.
 - Do a warmstart.
 - Halt the SDF.
5. If the SDF was loaded correctly, the following message appears at the system console:


```
SOFTWARE DUMP FACILITY (VER xxx.xxx/xxx)
```

The system executes the last instruction in the SDFCOM file, usually a HALT.
6. Check to see that the serial storage medium is on line and ready. Press the RUN key on the System Control Panel or enter RUN in response to the CMP prompt. For complete details on SDF commands, refer to the MPE V System Operation and Resource Management Reference Manual (32033-90005).

Memory Dump on Series 64/68

NOTE

Create a backup copy of the Software Dump Facility when the system is initially configured, and whenever you receive a software update. Then you will have it when you need it. Refer to the MPE V System Operation and Resource Management Reference Manual (32033-90005) for instructions.

1. Mount a magnetic tape on a logical device of device class DDUMP, then place the device on line.
2. If the C> or M> prompt does not appear on the console, press B^C. When the prompt appears, type:

DUMP [= i,c,d]

where *i* is the IMB number (0,1 or 2), *c* is the channel number (1-15), and *d* is the device number (0-7). All are decimal, and must be specified. Default values appear in the console banner. DUMP can be abbreviated DU.

The new values become the default values next time a dump is taken. If there is a powerfail before the next dump, the factory set default values (0,2,1), will take effect.

3. The SDF loads from the specified device and begins a serial execution of the SDF command file SDFCOM.
4. If the SDF loads correctly, the following message appears on the system console:


```
* * * SOFTWARE DUMP FACILITY (VER xxx.xxx/xxx)
      HALT
```
5. Check to see that the serial storage medium is on line and ready, then enter RUN in response to the DCU prompt on the console.

DUMP ANALYSIS OPTIONS

Four dump analysis options are available: one interactive, and three batch. Let your Hewlett-Packard System Engineer help you determine when a dump should be taken, and which analysis options you should use. The memory dump tape must be "write-enabled" when running DPAN5.

The four options are:

1. :RUN DPAN5.PUB.SYS [, EIGHTLPI] formats a dump (optionally at eight lines per inch), reporting on all data segments in memory, all MPE-resident data structures (formatted tables), the stack and associated segments of the current process at dump time, and stacks of other active processes in memory at the time of failure. If DPAN5 is run on the same system on which the dump was taken, the following three files are appended to the memory dump tape (if they have not been appended previously):
 - LOADMAP.PUB.SYS
 - CONFDATA.PUB.SYS
 - MPECHECK.PUB.SYS
2. :STREAM DUMPJOB.PUB.SYS generates a standard dump, a load map, the I/O configuration, and appends LOADMAP, CONFDATA, and MPECHECK to the end of the memory dump tape.
3. :STREAM DUMPMINI.PUB.SYS appends LOADMAP, CONFDATA, and MPECHECK to the dump tape, and generates a minidump. The minidump includes all MPE formatted tables, the stack, and segments associated with the processes which were current when the dump was taken (if any) the load map, and the I/O configuration.

The minidump reduces the amount of paper required to print a dump from a large memory system. However, it does not format all information available on the dump tape. For complete analysis of certain problems, it may be necessary to reprocess the dump tape to generate a standard dump.

4. :STREAM DUMPARCH.PUB.SYS appends LOADMAP, CONFDATA, and MPECHECK to the dump tape, but produces no listing. This is for archival of the dump for possible transport to another system for analysis.

OBTAINING A DPAN5 LISTING INTERACTIVELY

Follow these steps:

1. Restart the system with the COLDSTART, COOLSTART, WARMSTART, or RELOAD options.
- 2a. To get a full dump, log on to the system, then type:

```
:RUN DPAN5.PUB.SYS [ ,EIGHTLPI ]
```

DPAN5 first attempts to open an input file named MDUMP in the logon group. If unsuccessful, DPAN5 attempts to open MDUMP.PUB.SYS. If also unsuccessful, DPAN5 opens a file whose formal file designator is MDUMP, with the default device class name TAPE. If you prefer serial disc over tape, set up a file equation before running DPAN5, with formal designator MDUMP (e.g. :FILE MDUMP; DEV=SDISC). If you wish to use cartridge tape, the file equation must reflect the configured device class name of the cartridge tape. When DPAN5 is run in batch mode, DPAN5 directs its output to a file whose formal designator is DPANLIST, with device class name LP.

DPAN5 sends a tape mount request to the console when it is ready to read the MDUMP medium (magnetic tape, cartridge tape, or serial disc), then produces the formatted listing.

- 2b. If you want a formatted listing of a few specific system tables, rather than a listing of the entire memory dump, you can initiate the interactive dialogue with DPAN5. Run DPAN5 by entering:

```
:RUN DPAN5.PUB.SYS;PARM=10
```

The system will respond:

```
DPAN5 VER G.00.00 (C) HEWLETT-PACKARD CO. 1980
```

A tape request will appear on the System Console. When the tape is mounted, DPAN5 will read the dump, then prompt you for options. Levels of detail are indicated by indentation. Reply YES to any prompt to get to the next level of detail. At the deepest level, reply YES to select that option. Reply NO to any prompt to reject the option and skip to the next prompt at the same level. Reply ALL to any prompt to select all options at deeper levels and skip to the next prompt at the same level. The prompt "WHICH BANKS?" requests a list of banks for DPAN5 to print.

DPAN5 will ask:

```
MINIDUMP? YES or NO
```

If the minidump is invoked this way, you can request dump data on additional data segments by data segment number. The responses to "FORMAT TABLES?" and "PRINT PRIMARY MEMORY?" supplement the minidump option. For example, if minidump is specified, it is still possible to dump code segments in the PRINT PRIMARY MEMORY section. Or you may choose to dump only selected system tables as specified by the responses to the "FORMAT TABLES" dialogue. A sample dialogue follows:

```
WHICH DST (DECIMAL)?      Enter a 4-digit DST# in decimal or in octal
                           (%nnn), or press (RETURN) to move to the next
                           question level.

FORMAT TABLES?          YES, NO, or ALL; (RETURN) = NO.
  REGISTERS?             ALL means to automatically specify YES to all
  PROCESS CONTROL?      items in the next sublevel.
    CST?
    DST?
    PCB?
    STACK MARKERS?
    INTERRUPT CONTROL STACK?
  MEMORY MANAGEMENT?
    AVAILABLE REGION LIST?
    SEGMENT LOCALITY LIST?
    VDS PAGE ALLOCATION?
    VIRTUAL DISK SPACE BIT MAP?
  I/O MANAGEMENT?
    DRT?
    INTERRUPT LINKAGE?
    LOGICAL PHYSICAL DEVICE?
    DIT?
    DISC REQUEST TABLE?
    IOQ?
    SYSTEM BUFFERS?
```

```

    TERMINAL BUFFERS?
    TIMER REQUEST LIST?
    SIR?
    MONITOR?
    DISC CACHING TABLES?
    PRINT PRIMARY MEMORY?
    FORMATTED?
    CODE SEGMENTS?
    FREE AREAS?
    WHICH BANKS?      Enter decimal or octal number(s), separated by commas.
    
```

DPAN5 will now format and print the requested information.

3. Remove the tape from the drive.

EXAMPLES

Unless you requested specific information, the listing now contains: identification line, register page, code segment table (CST), code segment table extension (CSTX), data segment table (DST), process control block (PCB), monitor table, segment locality lists, available region lists, memory region tables, virtual disc space allocation information, virtual disc space bit map, device reference table (DRT), interrupt linkage table, logical/physical device table (LPDT), device information tables (DIT), input/output request table, disc request table, system buffer information, terminal buffer contents, timer request list, main memory contents (in octal), index, formatted stack information, formatted interrupt control stack (ICS), disc caching tables, and system internal resource table (SIR).

If you requested a minidump, the listing does not include the contents of main memory, but does include data on the current stack and segments. The example output shows some of the table headings you may expect to see in your output. The data should begin with a Hewlett-Packard heading at the top of the page, followed by the boxed Register Table (if you answered YES to this question). The last thing in the output is the DUMP INDEX. What is between these two depends upon what you have requested, but can include any of the table headings listed in this example:

```

HP3000 MEMORY DUMPG.00.00 OF SYS VER G UPDATE 00 FIX 00 DUMP TIME 3/14/84,5:40PM
(C) HEWLETT-PACKARD CO. 1980
          ***** REGISTERS *****
*****
* DATA SEGMENT      * CODE SEGMENT      * STATUS=103113      * ISR=140017 SERIES 44 *
*****
* DB BANK = 000012 * PB      = 144254 * MODE              = PRIV * RUN/HALT      = HALT      *
* DB      = 045230 * P        = 153226 * INTERRUPTS      = OFF  * IRQ              = OFF      ***
* S BANK  = 000012 * PL      = 167637 * TRAPS            = OFF  * CSRQ             = OFF      ***
* DL      = 045100 * PBBANK= 000000 * STACK OP        = LEFT * PARITY          = OFF      ***
* Q       = 077426 * (P-PB)= 006752 * OVERFLOW        = OFF  * POWERFAIL       = OFF      *
* S       = 077436 *              * CARRY           = ON   * POWERON         = OFF      *
* Z       = 100363 *              * COND CODE      = CCE  * DISP FLAG       = OFF      *
*              *              * SEGMENT #       = 113P * ICS FLAG        = OFF      *
*****
    
```

***** FIXED LOW MEMORY *****

(ADDR %0) CODE SEGMENT TABLE POINTER 006140
 (ADDR %1) EXTENDED CODE SEGMENT TABLE POINTER 007724
 (ADDR %2) DATA SEGMENT TABLE POINTER 002140

***** CST TABLE *****

SEGMENT NUMBER	SEGMENT NAME	MODE	REFERENCE BIT	TRACE	SEGMENT LENGTH	ABSOLUTE ADDRESS	***
-----	-----	----	-----	-----	-----	-----	---

***** EXTENDED CST TABLE *****

SEGMENT NUMBER	CSTBLK/PROCESS INDX	MODE	REFERENCE BIT	TRACE	SEGMENT LENGTH	ABSOLUTE ADDRESS	***
-----	-----	----	-----	-----	-----	-----	---

***** PROCESS CONTROL BLOCK (1ST HALF) *****

***** PROCESS CONTROL BLOCK (2ND HALF) *****

***** SIR TABLE *****

***** DUMP INDEX *****

NAME	PAGE #	FORMATTED	***
CODE SEGMENT TABLE		3	
DATA SEGMENT TABLE			
PROCESS CONTROL BLOCK		8	
CST EXTENSION		6	
SYSTEM GLOBAL AREA			
FIXED LOW CORE			
INTERRUPT CONTROL STACK			
SYSTEM BUFFERS		16	
UCOP REQUEST QUEUE			

DPAN4/DPAN5 ERROR MESSAGES

DPAN5 terminates after encountering an error. After any error, restream the job.

FWRITE failure in write to tape

FWRITE error temp file write failure

FREAD error in read from disc file

FCONTROL error in EOF write to tape

FSPACE error in movement of tape

FOPEN error - disc file open failed

FCLOSE error - temp file not closed

FCLOSE error - disc file not closed

File parity error - files invalid.

Dump may include several invalid file references in formatted table

Invalid dump tape - catastrophic tape errors - unable to read dump

Tape incorrectly prepared

You may also receive the message "Write ring absent". If so, rewind the tape, attach the write ring, and put the drive on line. Do not restream the job until corrective action is taken. This error is detected by MPE, not DPAN5.

SADUTIL performs emergency disc operations after the system has gone down. SADUTIL can store files on tape even if a system failure corrupts the system file directory. RECOVER5 reloads these files to disc. Because SADUTIL is a stand-alone program that runs without operating system control, no special MPE capability is required to run it, but SADUTIL is typically used by System Managers and System Supervisors.

SADUTIL is run after the system is shut down, and must be loaded from a tape prepared in advance. This preparation procedure is described in "PREPARING A SADUTIL COLD-LOAD MEDIUM" in this section. Once the system is halted, SADUTIL must be loaded and configured before it can be used. This is detailed in "LOADING AND RUNNING SADUTIL". Once SADUTIL is running, the following commands are available:

CAUTION

SADUTIL does not run under the safeguards of MPE, so careless use can corrupt the operating system. Use SADUTIL only in emergencies, or after the entire system has been backed up.

PDSK	Prints an octal or ASCII dump of any given area of a specified disc volume.
PDTT	Prints the defective track table of a specified disc volume.
PFIL	Prints descriptions of files contained in the system file directory.
PVOL	Prints information contained in the volume label of a specified disc volume.
EDIT	Modifies the contents of a disc volume.
FIND	Searches a system disc for file labels.
SAVE	Retrieves files from disc and copies them to magnetic tape. Can't be used with private volumes.
COPY	Copies the contents of one disc pack to another.
OUTM	Sets the output mode of print functions to printer output or console output.
CLID	Sets all cold-load IDs to 1.
CONF	Initiates the device configuration dialogue for additional devices.
HELP	Offers an explanation of all SADUTIL commands.
STOP	Terminates the SADUTIL program.

PREPARING A SADUTIL COLD-LOAD MEDIUM

SADUTIL is designed to recover files after a system failure so it must be stored on an external medium before it is needed. Use the program COPYDUS to create the cold-load medium. The storage medium can be either a flexible disc, a reel tape, or a cartridge tape. If you use flexible disc or cartridge tape, remember to format and serialize it before you run COPYDUS.

You must have the following files to create a copy of the Diagnostic Utility System (DUS) on flexible disc, magnetic tape, or cartridge tape:

COPYDUS.HP32231.SUPPORT	(Program File)
FLOPDUS.HP32231.SUPPORT	(Flexible Disc)
TAPEDUS.HP32231.SUPPORT	(Magnetic Tape)
CARTDUS.HP32231.SUPPORT	(Cartridge Tape)

1. To create a DUS medium, log on as follows:

```
:HELLO FIELD.SUPPORT,HP32231
```

2. Format and serialize your cartridge tape or flexible disc, using the >FORMAT and >SERIAL commands of the VINIT subsystem, as described in the MPE V System Operation and Resource Management Reference Manual (32033-90005).

3. Check to be sure that you have the required files available:

```
:LISTF @DUS.HP32231.SUPPORT
```

4. Run the COPYDUS program:

```
:RUN COPYDUS
```

COPYDUS identifies itself and asks for the identity of the medium on which you want to store the DUS:

```
DUS COPY ROUTINE REVISION 1.00  
ENTER MEDIA TYPE (FLOPPY DISC, CARTRIDGE TAPE, MAG TAPE):MAG TAPE
```

5. Depending upon how you answer the "MEDIA TYPE" question, COPYDUS will copy either FLOPDUS, TAPEDUS, or CARTDUS onto your formatted and serialized scratch medium. You are reminded, before the copy starts, to have clean scratch medium:

```
INSTALL MEDIA (MEDIA DOES NOT CONTAIN DUS FILE).
```

6. Mount the installation medium at this time. You should not mount the medium before the mount request, or after the mount reply.

7. The program identifies the medium you have mounted, and requests that you move to the System Console to continue the dialogue. When you respond to the tape request on the System Console, the following will appear:

```
BEGIN TRANSFER OF DATA.
BEGIN VERIFICATION OF DATA.
```

```
END OF PROGRAM
:
```

CAUTION

The latest version of SADUTIL is modified for MPE V/E. The MPE V/E version of the Diagnostic Utility System (DUS) tape will function correctly on an MPE IV (and MPE V/P) system, but the MPE IV (and MPE V/P) version will not work under MPE V/E. When you use SADUTIL, be sure that you have the correct version of the DUS tape.

LOADING AND RUNNING SADUTIL

SADUTIL is loaded as a program file under the Diagnostic Utility System (DUS). Before using SADUTIL, you must cold-load the DUS.

Cold-Loading the DUS on Series 39/40/42/44/48

1. Mount the medium containing the Diagnostic Utility System on the appropriate drive and place the drive on line, if applicable.
2. On the System Control Panel, set the LOAD thumbwheel switches to the channel address and device address of the drive containing the DUS.
3. From the System Control Panel, press HALT key, then press the LOAD key.

From the CMP, press RETURN. When the CMP prompt (->) appears on System Console, enter HALT. When another prompt is printed, enter LOAD.

The DUS is now read into memory, and the following message appears on the System Console:

```
DIAGNOSTIC/UTILITY SYSTEM    REVISION 00.00

ENTER YOUR PROGRAM NAME
```

4. Continue with the steps in "SADUTIL CONFIGURATION DIALOGUE".

Cold-Loading the DUS on Series 64/68

1. Mount the tape containing the DUS on the tape drive and place the drive on line.
2. If the DCU prompt (C>) is not already present on the System Console, press RETURN. When you see the prompt, enter HALT. When another prompt appears, enter LOAD.

If you wish to cold-load from a device other than the preset default device, enter the IMB number, channel number and the device number, separated by commas, after the LOAD:

```
C>LOAD imb,channel,device
```

NOTE

Once you have overridden the default cold load device in this way, the new values become the default until power is cycled on the system. At that time, the default values initially set by the factory take effect.

At this point the tape containing DUS is read into memory, and the following message appears on the System Console.

```
DIAGNOSTIC/UTILITY SYSTEM      REVISION 00.00
```

```
ENTER YOUR PROGRAM NAME
```

3. Continue with the steps in the "SADUTIL CONFIGURATION DIALOGUE".

SADUTIL Configuration Dialogue

When you reach the point in cold loading the DUS that you are asked for a program name, respond:

```
ENTER YOUR PROGRAM NAME  
SADUTIL
```

SADUTIL will identify itself and begin the interactive Configuration Dialogue:

```
DISC UTILITY C.03.03 (C) Hewlett-Packard Co., 1982
```

SADUTIL requires you to specify the configuration of any disc on which an operation is to be performed. Usually, this means all system discs configured when the system failed.

LDEV 1 must be assigned to the disc device which is LDEV 1 under the MPE operating system. Assign LDEV 2 through LDEV *n* to other discs, regardless of their LDEV assignments under MPE.

If you are using private volumes, configure the master disc of the private volume, instead of the system disc, as LDEV 1. Configure any private slave volumes as LDEV 2, LDEV 3, etc. If the private volumes are configured in this way, SADUTIL operates on the private volumes instead of the system volumes. SADUTIL cannot work with private and system volumes at the same time.

- | Step | Procedure |
|------|---|
| 1 | <p>LIST LOGICAL DEVICES? <u>Y</u></p> <p>To print a listing of all the logical devices, DRT numbers, unit numbers, types, and subtypes, currently in the LDEV Table, enter <u>YES</u> or <u>Y</u>. To skip the listing enter <u>NO</u> or <u>N</u> or press <u>RETURN</u> .</p> <p>The printer configuration is part of the Diagnostic Utility System (DUS). To access the DUS, use the <u>STOP</u> command to exit SADUTIL (thereby entering "MANAGER" mode). Execute the <u>LISTIO</u> command, then enter <u>CHANGEIO printer TO chnl,dev</u>. Finally, type <u>EXIT</u> to leave "MANAGER" mode and resume SADUTIL.</p> |
| 2a | <p>DISC CONFIGURATION CHANGES? <u>Y</u></p> <p>To change or add devices to the Logical Device Table, enter <u>YES</u> or <u>Y</u>. To skip and leave table unchanged, enter <u>NO</u> or <u>N</u> or press <u>RETURN</u> .</p> |
| 2b | <p>PRIVATE VOLUME SET? <u>Y</u></p> <p>To specify that you will be working with private volumes, respond <u>Y</u>.</p> |
| 2c | <p>LOGICAL DEVICE? <u>1</u></p> <p>Configure the master disc of the private volumes as LDEV 1. Configure any slave private volumes as LDEV 2, LDEV 3, etc. SADUTIL will operate on the private volumes and not on the system volumes. To add, delete, or change configuration, enter the logical device number (decimal). Press <u>RETURN</u> to go straight to Step 2h.</p> |
| 2d | <p>DRT? <u>49</u></p> <p>Enter hardware DRT number (decimal) of the disc to be referenced by this LDEV. Enter <u>0</u> to delete this LDEV and return to Step 2c.</p> |
| 2e | <p>UNIT? <u>0</u></p> <p>Enter hardware unit number (decimal) of the disc drive. This must be 0 for any type other than Type 0 or 3 discs.</p> |
| 2f | <p>TYPE? <u>3</u></p> <p>SELECT the disc type from the "SADUTIL DEVICE TYPES AND SUBTYPES" list, and enter it.</p> |
| 2g | <p>SUB-TYPE? <u>8</u></p> <p>Enter the subtype. Refer again to the "SADUTIL DEVICE TYPES AND SUBTYPES" descriptions. Be sure that the Type and Subtype entries are known to the MPE operating system (i.e. an HP 7906 disc may need to be assigned more than one LDEV# to reference its various logical parts).</p> <p>The program will continue to loop from here back to Step 2c until all discs in the current volume set are configured, and you press <u>RETURN</u> at Step 2c.</p> |

SADUTIL

- | Step | Procedure |
|------|---|
| 2h | <p>LIST LOGICAL DEVICES? <u>Y</u></p> <p>(<u>Y</u> or <u>YES</u>, or <u>N</u> or <u>NO</u> or <u>RETURN</u>.) This prints a listing of all logical devices, DRT numbers, unit numbers, types, and subtypes currently in the LDEV Table. LDEV 1, the system disc (or master disc, if you are using private volumes), must be configured by this point. If it is not, SADUTIL takes you back to Step 1.</p> |
| 3a | <p>SERIAL DEVICE CHANGES? <u>Y</u></p> <p>To modify the configuration of the magnetic tape or serial disc used by the >SAVE and >FIND commands, enter <u>Y</u> or <u>YES</u>. To leave unchanged and skip to Step 5, enter <u>N</u> or <u>NO</u> or <u>RETURN</u>.</p> |
| 3b | <p>DRT? <u>41</u></p> <p>Enter hardware DRT# (decimal) of new device.</p> |
| 3c | <p>UNIT? <u>0</u></p> <p>Enter the unit number of the new device.</p> |
| 3d | <p>TYPE? <u>24</u></p> <p>Configure any tape or serial disc that is indicated as serial in the SADUTIL Device Types and Subtypes description.</p> |
| 3e | <p>SUB-TYPE? <u>0</u></p> |
| 4 | <p>LIST SERIAL DEVICE? <u>Y</u></p> <p>To list the serial device configured in Steps 3a to 3d, enter <u>Y</u>.</p> |
| 5 | <p>ENTER FUNCTION?</p> <p>This ends the Configuration Dialogue. Enter one of the functions described on the following pages, or <u>STOP</u> to terminate SADUTIL. When you type <u>STOP</u>, control returns to the DUS.</p> |

SADUTIL Device Types and Subtypes

This summary consists of devices, types and subtypes recognized by the MPE operating system. Be sure that you use matching Types and Subtypes to describe your devices.

Device	Type	Subtype	Serial	Description
HP 7920A/S	0	8	*	Moving Head Disc
HP 925A/S	0	9	*	Moving Head Disc
HP 7906A	0	10	*	Removable Cartridge Only
		11		Fixed Platter Only
		12		Entire Drive
HP 9895	2	0	*	Flexible Disc Unit (Single)
		1	*	Flexible Disc Unit (Double)
HP 9110A	3	0	*	Cartridge Tape Unit
HP 7911A	3	1		Winchester Disc
HP 7912A	3	2		Winchester Disc
HP 7914A	3	2		Winchester Disc
HP 7933A	3	8	*	Moving Head Disc
HP 7935H	3	8	*	Moving Head Disc
HP 7970E	24	0	*	Magnetic Tape Drive
HP 7976A	24	1	*	High Speed Magnetic Tape

CLID

Sets all cold-load IDs to 1.

SYNTAX

CLID

OPERATION

If INITIAL is aborted during a COOLSTART or WARMSTART, and some cold-load IDs have been updated and others have not, the next attempt to use INITIAL will report that the volume table is corrupt and that a RELOAD is necessary. Actually, the only problem is that the CLIDs do not match. In that case, use the CLID command to reset the cold-load IDs to 1, to avoid performing a RELOAD.

EXAMPLE

ENTER FUNCTION: CLID

WARNING!! This function will rewrite all Cold Load ID's
Are all system domain volumes mounted and ready? Y

Cold load ID's written in system tables
Cold load ID written on LDEV#1
Cold load ID written on LDEV#2

ENTER FUNCTION:

Initiates the device Configuration Dialogue for the configuration of additional devices.

SYNTAX

```
CONF [ldev]
```

PARAMETERS

ldev A logical device number to be configured. Omit this parameter if you wish to be prompted for the device(s) to configure.

OPERATION

If you forgot to configure a device during the configuration dialogue, use CONF to return to that phase now. Refer to the "SADUTIL CONFIGURATION DIALOGUE" at the beginning of this section.

If *ldev* is specified, only that device will be configured, and the dialogue will begin at Step 2d. If no *ldev* is specified, the dialogue will start at Step 2a, and you can configure any device number.

EXAMPLE

To configure LDEV 6 (after the initial SADUTIL configuration dialogue has been completed):

```
ENTER FUNCTION:  CONF 6
DRT? 32
UNIT? 0
TYPE? 0
SUBTYPE? 10
LIST LOGICAL DEVICES? Y
```

LDEV	DRT	UNIT	TYPE	SUBTYPE
----	----	----	----	-----
1	15	0	0	11
2	15	0	0	10
6	32	0	0	10

```
ENTER FUNCTION:
```

COPY

Copies the contents of one disc pack to another.

SYNTAX

COPY

OPERATION

CAUTION

COPY is not a supported method of system backup.

If discs are of different types, the size of the smaller disc determines the amount of data copied. Both disc packs must have defective tracks reassigned. Packs with deleted tracks are not copied.

Discs of different types will always have the defective track table copied. If the discs are the same type, the defective track table is handled as follows:

- If the "from" disc has not been initialized by MPE, the Defective Track Table is copied to the "to" disc.
- If the "from" disc was initialized by MPE and the "to" disc was not, the defective track table is copied but all entries are deleted.
- If both the "from" and "to" discs are MPE-initialized, the defective track table will not be copied.

The COPY function informs you if the disc is not MPE-initialized, or if the volume table information disagrees with the way you have configured the disc, by issuing the following messages:

```
*WARNING*   LDEV #nn NOT INITIALIZED
or
*WARNING*   LDEV #nn CONFIGURED SUBTYPE DOES NOT AGREE WITH VOLUME TABLE
```

You are given the option of continuing after a warning.

EXAMPLE

```
ENTER FUNCTION: COPY
FROM LOGICAL DEVICE? 1
TO LOGICAL DEVICE? 2
MOUNT SCRATCH PACK ON LDEV #2
Press <RETURN> when mounted.
ENTER FUNCTION:
```

Modifies the contents of a disc volume.

SYNTAX

EDIT

OPERATION

This function prompts you with ">". You may then request a disc modification by typing an EDIT command described below. If you respond to the prompt by pressing **RETURN**, SADUTIL exits from the EDIT function and prompts you for another function.

>OUTM {C}
{P}

Sets the output mode for PDSK to the console (C) or the line printer (P).

>DISC [*ldev*]

Specifies the logical device on which the volume to be edited exists. (The initial *ldev* is 1)

>BASE [*basesector*]

Specifies the sector address to which all disc address references in the PDSK and MODIFY commands are relative. (Default is 0.)

>PDSK

Prints an octal/ASCII dump of any given area on the volume specified by the >DISC command. (Same as PDSK function.)

>MODIFY [*discaddr*][,*wordloc*][,*wordcount*]

Lets you modify specified words on the disc volume. (Defaults: *discaddr* = 0, *wordloc* = 0, *wordcount* = 1) These parameters may be entered in octal if preceded by a "%" sign. SADUTIL responds to >MODIFY with:

```
SECTOR % sectoraddr
      wordloc : oldcontents : =
```

where:

sectoraddr is the absolute octal sector address, not relative to *basesector*.

wordloc is the octal location, within the sector, of the word to be modified.

oldcontents is the current octal value of the word to be modified.

The new contents of the word should be entered as a string of six or fewer octal characters. Enter / to terminate the >MODIFY function and be prompted for a new command.

EDIT repeats the above display for each word to be modified, and EDIT confirms each modification by displaying "WRITTEN" on the console.

EXAMPLES

To modify disc 2, sector 0 (relative to *baseaddr*), word 7, with printer output:

```
ENTER FUNCTION: EDIT
>DISC 2
>OUTM P
>BASE 0
>MODIFY 0,7,1
SECTOR % 0
  7: 012451:= RETURN
WRITTEN
```

To modify three words of sector 2 (relative to *baseaddr*) beginning at word 2:

```
>MODIFY 2,2,3
SECTOR % 1
  2: 000014:=000014
  3: 000004:=/
WRITTEN
```

To modify one word of sector 1 (relative to *baseaddr*) beginning at location 2:

```
>MODIFY 1,2      Third parameter is missing, default is 1.
SECTOR % 1
  2: 000015:=14      Contents changed.
WRITTEN
```

To modify three words of sector 0 (relative to *baseaddr*) beginning at location 126:

```
>MODIFY 0,126,3
SECTOR % 0

177: 000000:= 0      Data written only if new differs from old.
SECTOR % 1
  0: 000003:= 2      Contents not changed.
WRITTEN
```

(There are 128 words per sector, so the third word is in the next sector.)

FIND

Scans a system disc for file labels.

SYNTAX

```
FIND
```

OPERATION

FIND allows files to be saved even when system directory has been destroyed.

FIND operates very slowly; it is for emergencies only. The function expects file sets to be in the form *filename.groupname.accountname*. Any of those parameters can be replaced by the "wild card" character (@), to specify "all members of the set". This feature will save time when attempting to FIND several files.

Upon finding a file label, SADUTIL displays:

- The file name, group and account.
- The device number of the disc.
- The disc address of the file label.

EXAMPLE

```
ENTER FUNCTION: FIND
FIND SCANS FOR FILE LABELS
THERE IS NO GUARANTEE THAT THE LABELS OR FILES ARE INTACT
IF THE FILES ARE REQUESTED TO BE SAVED, MANY MAY BE BAD OR ALREADY PURGED
IF THE FILE INFO ENDS WITH '?????', THEN THE FILE IS EITHER ALREADY
  PURGED OR IT WAS LEFT OPEN WHEN THE SYSTEM FAILED
FILES LISTED WITH '?????' WILL NOT BE SAVED
AN ATTEMPT CAN BE MADE TO SAVE VIA SAVE BY LDEV # AND SECTOR ADDRESS

DO YOU WISH FILES FOUND TO BE SAVED? Y

ENTER LDEV NUMBER TO SCAN: 1
ENTER FILE SET TO FIND: @.RANDAZZO.MPEM      File subset specific.
  DATE? 1/1/83
SCANNING LDEV 1 FOR FILE LABELS

HIT CONTROL A OR Y OR BREAK TO STOP 'FIND'

PLACE SERIAL DEVICE ON LINE AND PREPARE IT FOR WRITE
SECTORS/TRACK=48
TOTAL TRACKS=4075
```

If you answer Y to the question "DO YOU WISH FILES FOUND TO BE SAVED?", FIND invokes the SAVE command. If you respond N, you must individually save each file that you want to keep with the SAVE command. FIND produces this listing on the device selected by OUTM :

SCANNING LDEV 1 FOR FILE LABELS

MODIFIED SINCE 1/ 1/83

LDEV= 1,DRT= 4,UNIT= 0,TYPE= 0,SUBTYPE= 8

FILENAME	CREATED	MODIFIED	ACCESSED	SECTOR
SL .PUB .SYS	11/ 1/72	5/ 3/83	8/ 3/83 %	15675 ?????
SL .PUB .SYS	WAS EITHER PURGED OR LEFT OPEN			
HIDTAPE0.PUB .SYS	8/ 3/81	6/ 6/83	8/ 1/83 %	34417
HIDTAPE0.PUB .SYS	WAS SAVED!			
CSDUMMY.PUB .SYS	9/23/81	6/ 6/83	7/ 1/83 %	34436
CSDUMMY .PUB .SYS	WAS SAVED!			
IOCDRDO.PUB .SYS	6/22/81	6/ 6/83	6/ 1/83 %	34444
IOCDRDO .PUB .SYS	WAS SAVED!			

Offers an explanation of all SADUTIL commands.

SYNTAX

HELP

EXAMPLE

ENTER FUNCTION: HELP

Below are all the commands supported by SADUTIL

CLID	Set all Cold Load ID's to 1
CONF [ldev]	Configure logical devices
COPY	Copy one disk to another
DEBUG	Enter symbolic debugger
EDIT	Below are the five edit commands
BASE [basesector]	Specifies base sector number
DISC [ldev]	Specifies logical device to edit
MODIFY [diskaddr][,wordloc][,wordcount]	Modify a sector
PDSK	Print Disk. Enter address as [address][,sectcount][;AIO]
OUTM [C P]	Output to Console or Printer
FIND	Scan a disk for file labels
HELP	Explain facility
OUTM [C P]	Output to Console or Printer
PDTT [ldev]	Print Defective Tracks Table
PDSK [ldev]	Print Disk. Enter address as [address][,sectcount][;AIO]
PFIL	Print file names from directory
PVOL [ldev]	Print Volume Label of ldev
SAVE	Save files to serial device
STOP	Exit SADUTIL

ENTER FUNCTION:

OUTM

Sets the output mode of print functions to printer or console.

SYNTAX

```
OUTM [C]  
     [P]
```

PARAMETERS

- C Directs output of print functions to the console.
- P Directs output to the line printer in 132-character lines.

OPERATION

The default output mode is C (the console).

EXAMPLE

To set the output mode to the line printer:

```
ENTER FUNCTION: OUTM P
```

To set the output mode to the console:

```
ENTER FUNCTION: OUTM
```

or

```
ENTER FUNCTION: OUTM C
```

Prints an octal or ASCII dump of any given area of a specified disc volume.

SYNTAX

```
PDSK [ldev]
```

PARAMETERS

ldev The logical device number of the disc volume.

OPERATION

PDSK repeatedly prompts you with:

ENTER ADDRESS:

You may respond with the following:

```
firstsect {,numsect } {;A[,0]}  
          {:lastsect } {;0[,A]}
```

where:

firstsect is the starting sector number to dump.

numsect is the number of sectors to be dumped, starting with *firstsect*.

lastsect is the last sector number to dump. Sector numbers *firstsect* through *lastsect* will be dumped.

A prints the dump in ASCII.

0 prints the dump in octal.

If you omit the A and 0 parameters, the dump is printed in whatever format you specified previously. The initial format is octal.

After the dump is printed, you are prompted again with "ENTER ADDRESS:".

To abort the dump, press Y^C. After aborting, PDSK asks you to enter another address. Press RETURN to terminate PDSK.

EXAMPLE

To print an octal dump of sector 0 of the volume residing on logical device 1:

```
ENTER FUNCTION: PDSK 1  
ENTER ADDRESS: 0  
ENTER ADDRESS: RETURN  
ENTER FUNCTION:
```

To print an ASCII dump of sectors 0 through 1 of the volume residing on logical device 1:

```
ENTER FUNCTION: PDSK 1  
ENTER ADDRESS: 0:1;A  
ENTER ADDRESS: RETURN  
ENTER FUNCTION:
```

PDTT

Prints the Defective Track Table of a specified disc volume.

SYNTAX

```
PDTT [ldev]
```

PARAMETERS

ldev The logical device number of the disc volume.

OPERATION

MPE records defective disc areas in the Defective Track Table of each volume. The HP 7911, 7912, 7914, 7933, and 7935 disc drives do not have defective track tables; do not use PDTT with them. The DTT is found in sector 1 (the second sector) in this format:

Word	Contents
0	Number of entries (<i>n</i>) in the table (ranges from 0 to 120).
1- <i>n</i>	(Bits 0:14) = Track number. (Bits 14:2) = Status, where: 0 = Suspect track. 1 = Suspect alternate track. 2 = Deleted track. 3 = Reassigned track.
121-125	Reserved (filled with zeros).
126	Next available alternate track (moving-head discs only).
127	Logical disc pack size (in cylinders for moving-head discs, or tracks for fixed-head discs).

EXAMPLE

To print the defective track table of the volume residing on logical device 1:

```
ENTER FUNCTION: PDTT 1
LDEV= 1, DRT= 15, UNIT= 0, TYPE= 0, SUBTYPE= 5
LOGICAL PACK SIZE = 400 CYLINDERS, 11 ALTERNATE TRACKS AVAILABLE
NO ENTRIES IN DTT
ENTER FUNCTION:
```

PFIL

Prints descriptions of files contained in the system file directory.

SYNTAX

PFIL

OPERATION

PFIL repeatedly prompts:

ENTER NAME:

User response is:

filename [*groupname* [*acctname*]] [,*detail*]

The parameters *filename*, *groupname*, and *acctname* can be replaced with "@" to signify "all members of the set." Default is all files in system.

The parameter *detail* can be 0 (default), 1, or 2, depending on the type of information desired:

- 0 Requests a display of file names only
- 1 Requests a display of the name, LDEV and sector address of the first extent (i.e., file label) in the designated group and the account.
- 2 Requests a display of the name, creation date, last modification date, and last access date of a file.

To abort the PFIL, press Y^C.

If SADUTIL encounters an invalid file label during a PFIL function, it prints an asterisk (*) immediately after the file name.

EXAMPLES

To determine if the file SPL.PUB.SYS exists on the system:

```
ENTER FUNCTION: PFIL
ENTER NAME: SPL.PUB.SYS

ACCOUNT = SYS        GROUP = PUB

SPL
```

To determine the date of creation, last modification, and last access for the file SPL.PUB.SYS :

ENTER FUNCTION: PFIL
ENTER NAME: SPL.PUB.SYS,2

ACCOUNT = SYS GROUP = PUB

SPL 8/29/74 12/30/75 1/29/76

To print all files in the SYS account:

ENTER FUNCTION: PFIL
ENTER NAME: @.@.SYS

To print the file names, logical device numbers, and beginning sector address for all the files:

ENTER FUNCTION: PFIL
ENTER NAME: @,1

PVOL

Prints information contained in the volume label of a specified disc volume.

SYNTAX

```
PVOL [ldev]
```

PARAMETERS

ldev The logical device number of the disc volume.

OPERATION

The volume label, located in the first sector (sector 0) of the disc, is written in this format:

Words	Contents
0-5	On the system disc, this field contains the bootstrap input/output program. On other discs, this field is filled with zeros.
6	(Bits 6:6) Disc type. (Bits 12:4) Disc subtype.
7	Cold load count, incremented each time the system is cold loaded.
8-9	The characters "3000", used to verify that the disc label is valid.
10-13	The volume name, left-justified and padded with blanks.
14-127	Reserved (initialized to zeros).

EXAMPLE

To print volume label information for LDEV 2:

```
ENTER FUNCTION: PVOL 2
LDEV= 2, DRT= 15, UNIT= 0, TYPE= 0, SUBTYPE= 4
TYPE= 0, SUBTYPE= 4, C-L ID= 5419 , VOL. ID= SSTV2
```

SAVE

Retrieves files from disc and copies them to magnetic tape. Can't be used with private volumes.

SYNTAX

SAVE

OPERATION

SAVE is used with the EDIT function (for editing discs) and the RECOVER5 program (for creating disc files) after a catastrophic system failure. SAVE retrieves files from disc and copies them to magnetic tape for later recovery. Before an emergency, you must serialize and format the appropriate medium; it is impossible after a system crash.

If the system directory has been destroyed, it may still be possible to SAVE files by using the SADUTIL FIND command to locate files and select those you wish to save. To use the SAVE command when the system directory is intact, proceed as follows:

Step	Procedure
------	-----------

1	ENTER FUNCTION: <u>SAVE</u>
---	-----------------------------

2	SADUTIL now instructs:
---	------------------------

READY SERIAL DEVICE FOR WRITE

In response to this prompt, mount the tape to which the disc files are to be copied. (SAVE goes to Step 3 while you do this.)

3	SADUTIL now prompts:
---	----------------------

FILE NAME (OR LDEV #,%SECTOR ADDRESS)?

Indicate the file you want copied to tape. The file must reside on a disc defined during the "SADUTIL CONFIGURATION DIALOGUE", or with the CONF function. Indicate a file in one of the following two formats:

(a) *filename* [*.groupname* [*.acctname*]]

The parameters *filename*, *groupname*, and *acctname* can be replaced by @ to signify "all members of the set." (e.g. @.@.MPE or @.UTIL.MPE or @.@.@)

SADUTIL locates the specified file by searching the system file directory. Before copying the file, SADUTIL proceeds to Step 4.

(b) *ldev*, *sectoraddress*

The parameter *ldev* is the logical device number of the disc. The *sectoraddress* is the file location.

Find the sector address using the PFIL function (*detail=1*) before using SAVE, or from output previously obtained with the :STORE;SHOW or :RESTORE;SHOW MPE commands. Use this format when you know that some system file directory information is invalid. If used, SADUTIL takes you to Step 5 now.

To terminate the SAVE function, press **RETURN** , and SADUTIL will display the ENTER FUNCTION: prompt.

4 If format (a) was used in Step 3, this prompt will appear:

DATE? mm/dd/yy

where:

m *m* is a two-digit number representing the month (e.g. 06 or June).

d *d* is a two-day number representing the day of the month.

y *y* is a two-digit number representing the last two digits of the year.

To omit the date specification and copy all files requested in Step 3, press **RETURN** .

This prompt allows you to restrict retrieval to only those files modified since a specified date. SADUTIL now copies the files requested that were modified on or after the date specified.

The following information is printed for all files copied: file name, group name, account name, logical device number as configured by SADUTIL, and disc sector address of file label.

If a single file was selected in Step 3, SAVE returns to Step 3 now. If @.@.@ was selected in Step 3, SAVE terminates here and SADUTIL asks you for a new function. SAVE continues to Step 5 in other cases.

If a file, group, or account specified cannot be located, SADUTIL prints a message to this effect before returning to Step 3.

5 If format (b) was selected in Step 3, SADUTIL prints the following information for the file, as found in the file label at the specified address: file name, group name, and account name, followed by "CONTENTS OF LABEL". Immediately after this information, SADUTIL prints the prompt:

DO YOU WISH TO RETRIEVE THIS FILE(Y/N)?

Respond Y (or YES) to copy the file, or N (or NO) to avoid copying the file). In either case, SADUTIL performs the requested operation and returns to Step 3.

SPECIAL NOTES ON SAVE

Data is copied to tape in blocks of 128 words. The first block contains the file label, and the remaining blocks contain data and any user labels. Each file is terminated by an end-of-file (EOF) mark. The tape is terminated by an additional EOF.

One tape file cannot span more than one reel. During the SAVE operation, the tape is back-spaced and the second EOF is written when the end of the tape is detected. SADUTIL prompts you to mount a new reel, and the file being written is copied in its entirety to this reel.

Disc sector 28, decimal words 13 and 14, must contain the valid directory size and base address, respectively, for the SAVE to be successful. If a disc error is detected when reading Sector 28, the following dialogue occurs after Step 1 in the above series of steps:

Step Procedure

1.1 CAN'T READ SECTOR 28 OF SYSTEM DISC. STARTING SECTOR OF DIRECTORY?

Enter the starting address of the system Directory.

1.2 NO. OF SECTORS IN DIRECTORY?

Enter the size of the System Directory, in sectors.

If a magnetic tape operation fails, SADUTIL performs the following operations:

1 SADUTIL prints one of the following messages:

```
UNIT WENT OFF/LINE
NOT READY INTERRUPT
TRANSFER ERROR
CMD REJECT
TAPE RUN AWAY
TIMING ERROR
TAPE PARITY ERROR
SERIAL DEVICE FAILURE ldev
```

2 SADUTIL then prints:

```
THE FILES ON THIS TAPE WILL BE SAVED.
PLACE BAD TAPE BACK ONLINE IF IT IS OFFLINE
HIT CR WHEN UNIT IS BACK ONLINE
MOUNT NEXT REEL
```

3 Now, SADUTIL continues to save the files.

Other error messages dealing with reading the disc (such as bad tracks; improper accesses; bad directory addresses or label entries; or files, groups, or accounts not located) may appear. These messages should be self explanatory. Errors resulting from sectors outside of the directory, absence of required account entries, bad tracks, or invalid label comparisons, can be investigated by using the EDIT function.

EXAMPLES

To retrieve the file SPL.PUB.SYS, use SAVE in one of the following two methods:

```
ENTER FUNCTION: SAVE
READY SERIAL DEVICE FOR WRITE
FILE NAME (OR LDEV#,%SECTOR ADDRESS)? SPL.PUB.SYS      By file name.
DATE? (RETURN)
SPL      .PUB          .SYS          1          %33560
```

or

```
FILE NAME (OR LDEV#,%SECTOR ADDRESS)? 1,%33560      Contents of label
SPL      .PUB          .SYS                          by disc address.
RETRIEVE THIS FILE(Y/N)? N
FILE NAME (OR LDEV#,%SECTOR ADDRESS)? (RETURN)
ENTER FUNCTION:
```

STOP

Terminates the SADUTIL program.

SYNTAX

STOP

SADUTIL ERROR MESSAGES

After an error message appears, SADUTIL continues operation and prompts for further input.

<p>*ACCOUNT NOT IN DIRECTORY*</p> <p style="margin-left: 40px;">{READ } DISC {WRITE} ERR ON {SEEK } LDEV# <i>ldev</i> STATUS=%<i>status</i> %<i>sectoraddr</i></p>	<p>In a PFIL function request, you referenced a file belonging to an account not present in the System File Directory. Re-enter the file reference with the proper account name.</p> <p>SADUTIL did not complete the input/output operation disc. The <i>ldev</i> is the logical device number of the volume, <i>status</i> is the hardware status word of the device after the operation, and <i>sectoraddr</i> identifies the disc sector where the error occurred. If the error occurred on a removable disc, remove the disc and replace it with a new scratch disc, if possible. Then re-run SADUTIL.</p>
<p>*FILE NOT IN DIRECTORY*</p>	<p>In a PFIL operation, you referenced a file not present in the System File Directory. However, the account and group names were valid and present. Re-enter a correct file name.</p>
<p>*FUNCTION NOT COMPLETED*</p>	<p>The operation was not completed. Another error message should accompany this.</p>
<p>*GROUP NOT IN DIRECTORY*</p>	<p>In a PFIL request, you referenced a file belonging to a group not present in the System File Directory. However, the account name was valid and present. Re-enter the file reference with the proper group name.</p>
<p>ILLEGAL DEVICE SPECIFICATION</p>	<p>You specified an input/output device for a function not requiring such a specification. Re-enter the function keyword correctly.</p>
<p>*INVALID*</p>	<p>Check syntax and spelling and try again.</p>
<p>*INVALID COMMAND*</p>	<p>Invalid EDIT command. Check spelling and syntax, and retry.</p>
<p>INVALID DEVICE SPECIFICATION</p>	<p>You entered the function name correctly, but not the input/output device. Check spelling, and try again.</p>
<p>INVALID DISC ADDRESS</p>	<p>With the EDIT function, you attempted to modify a nonexistent address on the disc. Try again, and enter a correct address.</p>
<p>LDEV <i>ldev</i> NOT DEFINED</p>	<p>If you forgot to configure <i>ldev</i> during the configuration dialogue, use the CONF function to do it now. Or, check your number, or specify a different logical device.</p>
<p>DRT #<i>nn</i> UNIT #<i>nn</i> NOT READY</p>	<p>The input/output function cannot be performed because the disc device is not ready (e.g. a removable cartridge has been removed). Get the disc ready, and try again.</p>
<p>MISSING DEVICE SPECIFICATION FOR <i>volumename</i> VOLUME</p>	<p>The System Volume Table contains <i>volumename</i>, but no such volume has been mounted and configured into the system. Configure the device.</p>

SYSTEM DISC NOT CONFIGURED

This message appears during the initial configuration dialogue when the system disc (LDEV 1) has not yet been configured. Proceed with the configuration.

VOLUME *volumename* FOR
LDEV *#nn* NOT IN
VOLUME TABLE.
ENTER VOLUME TABLE
ENTRY FOR LDEV:

This message appears when a configured disc is not present in the System Volume Table. The message only appears during those functions that require all volumes to be mounted (such as PFIL and SAVE). After you respond, SADUTIL continues with the function.

SADUTIL usually keeps track of volume names by reading the System Volume Table (SVT). But if the SVT is corrupt, SADUTIL asks you the volume table entry for each logical device. Remember that volume numbers are not always the same as logical device numbers. Your responses must correspond to the SVT at the time of the system failure.

INVALID SERIAL DEVICE

You are trying to configure a serial disc which has not been labeled as serial with the VINIT subsystem.

ILLEGAL COPY OPERATION

The specified destination is not a disc, or is the same as the source disc.

ERROR LDEV *#nn*
HAS DELETED TRACKS

SADUTIL has scanned the defective track table and found that logical device number *nn* has one or more deleted tracks.

WARNING REMAINING
SECTORS OF FROM DISC WILL
NOT BE COPIED

The source disc is larger than the destination disc, so excess data cannot be copied.

RECOVER5 is used in conjunction with SADUTIL to recover disc files after a serious system crash in which the directory is damaged. SADUTIL, which runs without MPE, will store selected files onto tape without using the directory. Once you have performed a RELOAD to restore the operating system and user files from a SYSDUMP tape, RECOVER5 can be used to restore the newer files saved by SADUTIL.

OPERATION

To recover files from a halted system, follow these steps:

1. Use the SAVE function of the SADUTIL utility program to put the necessary files on tape.
2. RELOAD the operating system and user files from your last full SYSDUMP tape (refer to the MPE V System Operation and Resource Management Reference Manual (32033-90005)).
3. If the previous accounting structure cannot be recreated during the RELOAD, create the accounts from the keyboard now. The MPE V System Operation and Resource Manual (32033-90005) also contains the instructions for creating new accounts.
4. Mount the tape or serial disc, prepared by SADUTIL, that contains the files to be copied back into the system.
5. If the files are on tape, put the tape drive online. If the files are on serial disc, use the file equation:

```
:FILE RECOVTP; DEV=SDISC
```

6. Log on and run RECOVER5 as follows:

```
:HELLO MANAGER.SYS  
:RUN RECOVER5.PUB.SYS
```

7. RECOVER5 identifies itself and asks you to enter the file sets to recover. Terminate your list with a (RETURN) in response to the > prompt.

You may enter file sets in the *file.group.account* format with the following "wild card" characters:

- @ Matches from zero to eight characters
- ? Matches any single character
- # Matches any single numeric character (0 to 9)

RECOVER2/RECOVER5

```
RECOVER5 G.00.00 (C) HEWLETT-PACKARD CO., 1983
ENTER FILESETS TO RECOVER
TERMINATE LIST WITH A NULL LINE
>P@.EGAN.MPEM
MORE?> ?SA?#@.TEST.ACCOUNT
MORE?> (RETURN)
```

8. Before recovering any files, RECOVER5 asks (but only once):

```
WISH TO KEEP EXISTING COPIES OF FILES?
Y (or N, as desired).
```

You must answer this question with one of the choices, or you will hang the program.

9. RECOVER5 begins copying the files, from serial disc or tape, to disc. When the program completes, it lists the names of all of the files that were restored and any error conditions encountered while they were being recovered.

10. You are now asked for other tapes:

```
IS THERE ANOTHER RECOVERY TAPE? Y (or N, as desired)
```

If there is another SADUTIL tape to restore, respond Y, mount the tape, and press (RETURN). Otherwise, RECOVER5 terminates.

11. When RECOVER5 terminates, you can restore any other files from partial backup tapes by using the :RESTORE command with the ;KEEP keyword parameter (refer to the MPE V System Operation and Resource Management Reference Manual (32033-90005)).

EXAMPLE

```
:HELLO MANAGER.SYS
:RUN RECOVER5.PUB.SYS
RECOVER5 G.00.00 (C) HEWLETT-PACKARD CO., 1983
ENTER FILESETS TO RECOVER
TERMINATE LIST WITH A NULL LINE
>P@.EGAN.MPEM
MORE?> ?SA?#@.TEST.ACCOUNT
MORE?> (RETURN)
WISH TO KEEP EXISTING COPIES OF FILES? (Y/N)Y
.
.
.
IS THERE ANOTHER RECOVERY TAPE? (Y/N)N
END OF PROGRAM
```

RECOVER2/RECOVER5 ERROR MESSAGES

The name of the current disc file being recovered is attached to the front of each of the messages which follow. Each RECOVER5 message is followed by a file system error message. For example:

```
WTFILE.DAZO.DATABASE - ERROR ON FOPEN OF EXISTING FILE - CODE 91
EXCLUSIVE VIOLATION: FILE ACCESSED EXCLUSIVELY (FSERR 91)
```

You should examine the file system error messages for additional information about why the error occurred. In the current example, someone tried to recover a file that was being exclusively accessed; the file must be closed by the present user before it may be purged. The RECOVER5 error message gave you a description of what occurred, and the file system error message told you why it happened.

DUPLICATE FILE (NOT LOADED)	You have tried to keep existing copies of files being recovered and a file with a duplicate name already exists. If you want to recover the file, purge the existing file or specify to RECOVER5 that you do not want to keep existing copies of files.
ERROR ON FOPEN OF EXISTING FILE - CODE <i>xxx</i>	You have tried to purge existing copies of files and RECOVER5 could not open the existing file to purge it. Note the file system error and take appropriate action.
PURGE ERROR - CODE <i>xxx</i>	You have attempted to purge existing copies of files. RECOVER5 was able to open the file but failed when attempting to close the file with the purge option. Note the file system error and take appropriate action.
FCLOSE FAILURE - CODE <i>xxx</i>	RECOVER5 encountered an error while attempting to close a file being saved with the FCLOSE SAVE option. Note the file system error, and take the required action.
LDEV DISC-ADDRESS	This is not an error; it indicates that the file was successfully recovered onto the system. RECOVER5 outputs the logical device number and absolute disc address of the file label after the file name, to indicate successful recovery of the file.
ATTACHIO ERROR: IOCB = % <i>xxxxxxxx</i>	An I/O error was detected when attempting to read or write the file label of the disc file being recovered. The <i>xxxxxx</i> is the ATTACHIO error status. Notify your support engineer.
DISC WRITE FAILURE - CODE <i>xxxx</i>	An error was reported by the file system when writing data to the disc file. Note the file system error and take appropriate action.
OUT OF DISC SPACE OR TOO MANY RECS ON TP	RECOVER5 discovered more data on the tape than was indicated in the file label. This indicates an internal problem with either RECOVER5 or SADUTIL. Notify your support engineer.
FOPEN FAILED - CODE <i>xxxx</i>	The FOPEN used to create the disc file to recover failed. Note the file system and take appropriate action.
FGETINFO FAILED	The FGETINFO call to obtain file information failed. This indicates a problem with RECOVER5 or MPE. Notify your support engineer.

RECOVER2/RECOVER5

The error messages which follow do not have file names attached to them. A "***" after the message indicates that a file system error message follows the RECOVER5 message.

MOUNT NEXT RECOVERY TAPE	RECOVER5 is finished with the current serial device medium. Mount a new one, and continue.
TAPE READ ERROR - CODE <i>xxxx</i>	A read error was detected when reading the first block of a file from a serial device. Note the file system error and contact your support engineer.
This version will work only on MPE-V	RECOVER5 will only run on MPE version G.00.00 or later. RECOVER2 runs on earlier versions.
FAILURE TO OPEN MAG. TAPE FILE **	The FOPEN of the serial device file RECOVTP encountered an error. Note the file system error, and take appropriate action.
CONTROL-Y DETECTED: ABORT OR CONTINUE?	You have entered <u>Y</u> ^C , and now have the option to continue or abort. Enter <u>C</u> to continue or <u>A</u> to abort.
IS THERE ANOTHER RECOVERY TAPE (Y/N)?	RECOVER5 is finished with the current serial device medium, and wants to know if there are more files to recover. If there is another tape or serial disc with data, mount it and continue processing.
CONTROL-Y WILL BE ACKNOWLEDGED AT THE END OF THIS FILE	You have entered <u>Y</u> ^C , and RECOVER5 is indicating that it will be acknowledged after the current file is recovered.

The following messages are output if errors are detected while the file sets are being parsed. A " ^ " is printed under the offending character. Note the error message, and re-enter all file sets.

ERROR BUILDING PATTERN IN PATTERNMATCH-NAME
ERROR BUILDING PATTERN IN PATTERNMATCH-GROUP
ERROR BUILDING PATTERN IN PATTERNMATCH-ACCOUNT
EXCESSIVE FILESETS IGNORED (10 ACCEPTED)
PART TOO LONG
ZERO LENGTH PART FOUND
FOUND WILDCARD AND EITHER \$ or *
FIRST CHARACTER OF PART MAY NOT BE NUMERIC
LOCKWORD MAY ONLY FOLLOW THE FILE PART
TOO MANY PERIODS WERE FOUND, THE MAXIMUM IS TWO
A * WAS FOUND, BUT IT WASN'T THE FIRST CHAR
BOTH \$ AND * MAY NOT BE SPECIFIED
A \$ WAS FOUND, BUT IT WASN'T THE FIRST CHARACTER
AN ILLEGAL CHARACTER WAS FOUND
NEEDED PARAMETERS TO THE PROC WERE MISSING

DISKED2/DISKED5

SECTION

XIII

The utility DISKED5 (DISKED5.PUB.SYS) runs on MPE V/E systems; DISKED2 (DISKED2.PUB.SYS) is the MPE V/P and MPE IV version. The dialog and operation of both are identical. Throughout this section, DISKED5 will be used in all examples and explanations.

DISKED5 allows you to modify or display the contents of a disc.

A regular user can operate in "file editor mode", modifying any word of the user's own file, but not a file label. A System Manager can operate in "disc editor mode", modifying any word on disc, or in file editor mode.

CAUTION

The normal safeguards of MPE are bypassed in privileged mode. When attempting to modify privileged data on disc, it is possible to destroy file integrity, or the MPE operating system itself. HP will investigate and attempt to resolve problems resulting from modification of privileged data, but this service is not included in the standard service contract. HP will not modify the MPE operating system to accommodate problems arising from your use of DISKED5.

OPERATION

1. To run DISKED5 type:

```
:RUN DISKED5.PUB.SYS
```

2. DISKED5 identifies itself and prompts you for an input:

```
DISKED5 G.00.00 (C) HEWLETT-PACKARD CO., 1983  
TYPE 'HELP' FOR INFO  
>
```

3. Respond to the ">" prompt with a DISKED5 command. Precede octal numbers with the % character (e.g. %34).
4. To terminate DISKED5, enter enter or EXIT in response to the > prompt.

COMMANDS

DISKED5 has ten commands; each will be described in detail in the following pages.

BASE	Sets the base sector number used with the >MODIFY and DUMP commands.
DEBUG	Calls the MPE DEBUG facility.
DISC	Specifies the logical device of the file to be modified.
DUMP	Displays selected disc sectors on a list device.
EXIT	Terminates DISKED5.
FILE	Activates the file editor mode, for System Managers.
HELP	Prints a summary of DISKED5 commands.
LIST	Specifies the output device for the >DUMP command.
MODIFY	Changes the contents of specified words on disc.
WIDTH	Selects wide or narrow format for output listings.

>BASE

Sets the base sector number used with the >DUMP and >MODIFY commands.

SYNTAX

```
>BASE absector
```

PARAMETERS

absector

A decimal or octal (prefixed with %) number specifying the absolute base sector to which the *relsector* parameter will be relative.

OPERATION

>BASE may be abbreviated >B.

>DEBUG

Calls the MPE DEBUG subsystem.

SYNTAX

```
>DEBUG
```

OPERATION

>DEBUG invokes the MPE DEBUG facility, described in the MPE Debug/Stack Dump Reference Manual (30000-90012). Do not abbreviate >DEBUG. You must have Privileged Mode (PM) to use DEBUG.

Type R to exit DEBUG and resume DISKED5.

Specifies the logical device of the disc to be modified.

SYNTAX

```
>DISC ldev
```

PARAMETERS

ldev The logical device number of the disc to be modified. The default is one.

OPERATION

This command may be used by System Managers only.

>DUMP

Displays selected disc sectors on a list device.

SYNTAX

```
>DUMP {relsector [,numsectors]} [,A]
      { ALL                }      
```

PARAMETERS

relsector The sector number, relative to the *absector* specified in the >BASE command, to use as the starting sector for the dump. The default is zero.

numsectors The number of sectors to dump. The default is one.

ALL Specifies that the entire disc is to be dumped.

A Specifies that the dump is to be in ASCII format.

OPERATION

This command dumps the selected sectors onto a list device. The starting sector is determined by adding *relsector* to the base sector specified in the *absector* parameter of the >BASE command.

The formal output file designator for DISKED5 is DEDILIST, with default device class LP. The output may be redirected using the >LIST command. Press Y^C to abort the output of the >DUMP command.

>DUMP may be abbreviated >D.

>EXIT

Terminates DISKED5.

SYNTAX

>EXIT

OPERATION

>EXIT may be abbreviated >E.

>FILE

Activates the file editor mode (System Managers only).

SYNTAX

```
>FILE filename
```

PARAMETERS

filename The name of the file to edit, in the form
filename[.groupname[.acctname]].

OPERATION

This command may be used by System Managers only. >FILE may be abbreviated >F.

Displays a summary of DISKED5 commands.

SYNTAX

```
>HELP
```

OPERATION

Do not abbreviate >HELP.

>LIST

Specifies the output device for the >DUMP command.

SYNTAX

```
>LIST {ldev }  
      {devclass}
```

PARAMETERS

ldev The logical device number where the listings from the >DUMP command should appear.

devclass The device class where the listings from the >DUMP command should appear.

OPERATION

The default list device is class LP. The formal output file designator is DEDILIST. >LIST may be abbreviated >L.

Changes the contents of the specified words on disc.

SYNTAX

```
>MODIFY sectornum, relwordaddr [, numwords]
```

PARAMETERS

- sectornum* A decimal or octal (prefixed with %) number, indicating the absolute sector address of the sector to modify. This value is added to the *absector* value specified in the >BASE command, if any, to determine the actual sector address to be modified.
- relwordaddr* A decimal or octal (prefixed with %) number, indicating the address of the first word to modify, relative to the beginning of the sector.
- numwords* The total number of words to be modified. The default is one.

OPERATION

>MODIFY changes the contents of the specified words on the disc. The address you actually modify is calculated by adding *absector* from the >BASE command to the *sectornum* specified in >MODIFY.

The >MODIFY command tells you the sector number of each word modified, and prompts you for contents with a comma (,):

```
SECTOR % sectornum LDEV=%ldev  
relwordaddr; wordcontents,
```

where: *sectornum* is the absolute sector address, *ldev* is the logical device number, *relwordaddr* is the address of the word to be modified, and *wordcontents* is the present contents of the specified word.

Enter a new value for the word, or enter an asterisk (*) or press **RETURN** to retain the old value. Inputs are assumed to be octal unless prefixed with a "#" (#4082), which indicates decimal. Input can be ASCII if specified with an apostrophe ('). ASCII input can be directed to either the left byte only, or to both bytes of the word. (e.g. 'f puts an f in the left byte, and 'wb puts w in the left byte and b in the right byte.

Successful changes are confirmed by the message "WRITTEN".

Enter / to abort the >MODIFY. Enter Y^C to abort the listing of the >MODIFY output.

>MODIFY may be abbreviated >M.

>WIDTH

Selects wide or narrow format for output listings.

SYNTAX

```
>WIDTH
```

OPERATION

>WIDTH generates the prompt "NARROW FORMAT?". Respond Y to allow space for ASCII characters to appear on the listing, or N for wide format.

>WIDTH may be abbreviated >W.

EXAMPLES

The output in the examples on this page show how changing the Base Address changes the sector being dumped, even though the >DUMP command parameter remains the same:

```
:RUN DISKED5.PUB.SYS
```

```
DISKED5 G.00.00 (C) HEWLETT-PACKARD CO., 1983
TYPE 'HELP' FOR INFO
```

```
>DISC 1          Select the System Disc.
>DUMP 0         Dump sector 0 to the system list device.
```

```
SECTOR %00000000000          LDEV = %0000001
000: 051531 051524 042515 020104 044523 041440 000011 021632 SY
010: 031460 030060 054517 042101 030440 020040 000000 000000 30
.
.
.
150: 000000 000000 000000 000000 000000 000000 000000 000000 ..
160: 000000 000000 000000 000000 000000 000000 000000 000000 ..
170: 000000 177777 062706 177777 000000 000620 000000 000626 ..
```

```
>BASE %10       The base sector number is reset to 10, so that when you
>DUMP 0         dump 0, as in the example above, 0 is now relative to a
                base of 10 (octal). The address 000 on the listing now
                shows the contents formerly listed as at address 010.
```

```
SECTOR %00000000010          LDEV = %0000001
000: 022000 141515 041707 022004 141603 033436 140007 041707 $.
010: 022016 141603 033437 140002 030371 140010 004500 022003 $.
.
.
.
150: 000600 031024 004300 014300 017702 140002 000106 025401 ..
160: 020320 051401 040053 020321 000300 020320 051402 040047 .
170: 021004 003243 020321 040044 021001 003243 020321 040041 ".
```

```
>EXIT
END OF PROGRAM
:
```

DISKED2/DISKED5

This example shows how you can >DUMP a sector without knowing the sector address, if you know the name of a file within the sector.

```
>DISC 1
>FILE DAN    Access an area by filename, rather than sector number.
>DUMP 0      You can then use DISKED5 commands which require sector numbers.
```

```
LOGICAL SECTOR 0          *** FILE LABEL ***
SECTOR %00000023600      LDEV = %000001
000: 042101 047040 020040 020040 046520 042525 052111 046040 DA
010: 046501 050120 020040 020040 041122 042516 042101 020040 MA
020: 020040 020040 020040 020040 020202 004040 000001 124046
030: 124046 124046 000000 000000 010403 000000 000000 000012 .&
040: 000731 000050 062746 021632 000001 177400 000200 000400 ..
050: 000013 000013 000000 000000 000400 023600 000000 000000 ..
.
.
.
150: 000000 000000 000000 000000 007070 025000 124046 000000 ..
160: 000000 000000 000000 000000 000000 000000 007424 001401 ..
170: 000000 000000 000000 000000 030415 006502 015502 000001 ..
>EXIT
```

```
END OF PROGRAM
:
```

This example shows the ASCII dump format. When you use the >DUMP command with the A parameter, the output is in ASCII, only. You may output octal with ASCII by using the >WIDTH command:

```
>DISC 1
>DUMP 0,1,A    Dump starting at sector 0, for one sector, in ASCII
```

```
SECTOR %00000000000      LDEV = %000001
000: SYSTEM DISC ..#.3000Y0DA1 .....
100: .....
200: .....
300: .....e.....
>EXIT
```

```
END OF PROGRAM
:
```

>DEBUG can be used to isolate a problem, >DUMP used to confirm the location of the errant values, and >MODIFY to change them.

>DEBUG

DEBUG PRIV.0.1000

.
.
.

?RESUME

Resume DISKED5 directly from DEBUG.

>DUMP %300

Dump the relative sector you are interested in changing.

```
SECTOR %00000000300      LDEV = %000001
000: 000000 000000 000000 000000 000000 000000 000000 000000 ..
010: 000000 000000 000000 000000 000000 000000 000000 000000 ..
020: 000000 000000 000000 000000 000000 000000 000000 000000 ..
030: 000000 000000 000000 000000 000000 000000 000000 000000 ..
040: 000000 000000 000000 000000 000000 000000 000000 000000 ..
.
.
.
```

>MODIFY %300,%20,2

Change the contents of the two words beginning at
at relative octal address 20.

SECTOR %00000000300 LDEV = %000001 Absolute sector number.

020: %000000,1 ← Present contents, and new contents.

021: %000000,2

WRITTEN

Changes are confirmed by DISKED5.

>DUMP %300

Verify change in dump.

```
SECTOR %00000000300      LDEV = %000001
000: 000000 000000 000000 000000 000000 000000 000000 000000 ..
010: 000000 000000 000000 000000 000000 000000 000000 000000 ..
020: 000001 000002 000000 000000 000000 000000 000000 000000 ..
030: 000000 000000 000000 000000 000000 000000 000000 000000 ..
.
.
.
```

DISKED2/DISKED5

These two examples show the difference between the wide and narrow formats. Note the beginning of ASCII at the end of the octal output. The ASCII output can be obtained without the octal by using the >DUMP command:

DISKED5 G.00.00 (C) HEWLETT-PACKARD CO., 1983
TYPE 'HELP' FOR INFO

>DUMP 0

```

SECTOR %00000000000          LDEV = %000001
000: 051531 051524 042515 020104 044523 041440 000011 021632 SY
010: 031460 030060 054517 042101 030440 020040 000000 000000 30
020: 000000 000000 000000 000000 000000 000020 003416 000000 ..
030: 000000 000000 000000 000000 000000 000000 000000 000000 ..
.
.
.

```

>WIDTH
NARROW FORMAT?

YES

>DUMP 0

```

SECTOR %00000000000          LDEV = %000001
000: 051531 051524 042515 020104 044523 041440 000011 021632
010: 031460 030060 054517 042101 030440 020040 000000 000000
020: 000000 000000 000000 000000 000000 000020 003416 000000
030: 000000 000000 000000 000000 000000 000000 000000 000000
.
.
.

```

Two ways to specify the output device for your dump are:

```

>LIST 6          LDEV # for Line Printer.
>DUMP 0
>LIST
>EXIT

```

and

```

>DISC 1
>LIST LP      Change the device class for a dump from the system default.
>DUMP 0
>EXIT

```

DISKED2/DISKED5 ERROR MESSAGES

**FWRITE ERR ON LIST <i>fchecknum</i>	DISKED5 output is incomplete. The <i>fchecknum</i> shows the error code returned by the FCHECK intrinsic. Refer to the MPE V Ininsics Reference Manual (32033-90007) for details.										
INVALID DISC ADDRESS	The >BASE address is invalid, or relative sector address plus absolute sector address in >DUMP is less than zero, or sector address plus absolute sector address in MODIFY is less than zero, or disc address in any of the three commands is too high for the disc being accessed.										
**IRRECOVERABLE DISC ERROR=%0000nn	Error during >DUMP or >MODIFY. %0000nn specifies the error: <table> <tbody> <tr> <td>%000004</td> <td>Invalid function.</td> </tr> <tr> <td>%000014</td> <td>Transmission error. CRC or track-specific error.</td> </tr> <tr> <td>%000034</td> <td>Transmission error.</td> </tr> <tr> <td>%000044</td> <td>SIO not ready.</td> </tr> <tr> <td>%000054</td> <td>Unit failure. All errors other than track-specific.</td> </tr> </tbody> </table> <p>After the disc error number is displayed, DISKED5 returns to prompt (>).</p>	%000004	Invalid function.	%000014	Transmission error. CRC or track-specific error.	%000034	Transmission error.	%000044	SIO not ready.	%000054	Unit failure. All errors other than track-specific.
%000004	Invalid function.										
%000014	Transmission error. CRC or track-specific error.										
%000034	Transmission error.										
%000044	SIO not ready.										
%000054	Unit failure. All errors other than track-specific.										
**WRITE ACCESS IS REQUIRED	You do not have the proper capability to access the file. See your System Manager.										
**SYS. MGR CAPABILITY IS REQUIRED	You need System Manager (SM) capability to perform this action.										
**SYS. MGR CAPABILITY REQUIRED TO MODIFY FILE LABEL	To modify a file, you must have legal access to it.										
**LOCAL DISC FILES ONLY	Only files on a local machine may be accessed by DISKED5. Files accessed via communications subsystems cannot be accessed by this program.										
**HARDWARE END OF FILE ENCOUNTERED	A hardware error was encountered in reading the input file.										
**PRIVILEGED MODE REQUIRED TO ENTER DEBUG	Privileged mode capability is required to enter DEBUG from DISKED5.										
DEVICE NOT DISC	A command referenced a device that is not a disc.										
UNABLE TO OPEN LIST DEVICE	DISKED5 could not access the referenced device.										
WRONG NUMBER OF PARAMETERS	An invalid number of parameters is specified for this command.										

DISKED2/DISKED5

FIRST PARAMETER INVALID
or
SECOND PARAMETER INVALID
or
THIRD PARAMETER INVALID

The parameter in the position indicated is invalid.

The purpose of **PATCH** is to give you access to the object code of a program file, and to let you display or modify it without recompiling. You can use **PATCH** to make simple changes to program instructions or to the variables in the global stack area. No special MPE capabilities are required to run **PATCH**.

CAUTION

PATCH can bypass normal MPE safeguards and modify the contents of privileged program files. It is possible, therefore, to corrupt a file or the entire MPE operating system. HP will investigate and attempt to resolve problems resulting from modification of privileged software, but this service is not included in the standard service contract. HP will not modify the MPE operating system to accommodate problems arising from your use of **PATCH** on privileged files.

The System Manager must prevent unauthorized use of **PATCH**, perhaps by placing a lockword on the program file.

OPERATION

In order to use **PATCH**, it is necessary to know the memory location where your program symbols are stored, the beginning locations of each program unit, and the offsets from these locations of each line of code. This information is obtained in several ways, depending on the source language of your program.

A FORTRAN program should be compiled with the **MAP**, **LABEL**, and **LOCATION** parameters in the **\$CONTROL** command. The **MAP** parameter generates a symbol map that lists all of the symbolic names in your program and the location in the data stack where the data is stored for each symbol. The **LABEL** parameter provides a label map which shows the offset of each labeled statement. The **LOCATION** parameter provides the offset of every statement as part of the source listing.

The COBOL and SPL programs to be patched, should be prepared with the **PMAP** parameter. The material which follows tells you how to prepare your program using the **PMAP** parameter in the **:PREP** command.

PATCH

Using PMAP

1. To use the PMAP parameter with the :PREP command, enter the following:

```
:PREP $OLDPASS,$NEWPASS;PMAP
```

The :PREP command and its parameters are described in the MPE V Commands Reference Manual (32033-90006). The USL file name is \$OLDPASS. The use of User Subprogram Library files for compiler output is described in the MPE Segmenter Reference Manual (30000-90011). The program file name is \$NEWPASS.

The use of PMAP in connection with the PATCH program is described below. A more general description of PMAP can be found in the MPE Segmenter Reference Manual (30000-90011), and in the MPE Debug/Stack Dump Reference Manual (30000-90012).

2. The PMAP parameter produces information in the following format:

```
PROGRAM FILE $NEWPASS.MPEVB3.KSE
```

```
SEG'          0
NAME          STT  CODE ENTRY SEG
DB'           1    0    0
FOPEN         3
PRINTFILEINFO 4

FREEDSEG      25
TERMINATE'    26
QUIT'IT       2 2557    2557
FCHECK        27
FERRMSG       30
TERMINATE     31
SEGMENT LENGTH      3204

PRIMARY DB     45  INITIAL STACK 2260  CAPABILITY      600
SECONDARY DB  770  INITIAL DL     0    TOTAL CODE      3204
TOTAL DB     1035  MAXIMUM DATA  ?    TOTAL RECORDS   27
ELAPSED TIME 00:00:01.700      PROCESSOR TIME 00:00.436
```

```
END OF PREPARE
```

```
:
```

The information has the following meaning:

SEG'	Segment name.
0	Logical segment number.
NAME	Program unit entry point name or external procedure name.
STT	Assigned entry number in Segment Transfer Table (STT).
CODE	Beginning location of procedure code in segment.

ENTRY	Location of entry point in this segment.
SEG	Logical segment number; if "?", the procedure is external to the segment, and external to the program file.
PRIMARY DB	The number of words, in octal, used for global stack variables and indirect references to array variables.
INITIAL STACK	The number of words (octal) in the stack when the program first runs. This initial size can increase up to the value specified in the MAXDATA parameter of the :PREP command.
CAPABILITY	Capability of program file (i.e. IA, BA, PM, etc.). Refer to Word 2 of UCAP in the Job Information Table (JIT), in the MPE V System Tables Reference Manual (32033-90010) for the translation of bit position to Capability.
SECONDARY DB	The number of words, in octal, used for array elements.
INITIAL DL	The number of words, in octal, that will be allocated to the Data Limit (DL) area of the stack, as specified in the DL parameter of the :PREP command.
TOTAL CODE	Total number of words, in octal, for all of the segments in the program file.
TOTAL DB	The sum of the Primary and Secondary DB areas.
MAXIMUM DATA	The number of words, in octal, specified in the MAXDATA parameter of the :PREP command. A "?" in this field means that the USL file was prepared with no maximum data specified. The value defaults to the configured system maximum.

Running PATCH

1. To run PATCH, type:

```
:RUN PATCH.PUB.SYS
```

2. PATCH identifies itself, then prompts you with "?" for the name of the file to list or modify:

```
PROGRAM PATCH G.00.00 (C) HEWLETT-PACKARD CO., 1976
FILE=?
```

3. Enter a one- to eight-character alphanumeric file name of the file whose code you wish to display or change. An invalid file name terminates PATCH with the following error message:

```
*** ERROR *** UNABLE TO OPEN FILE
```

PATCH

4. When your file name is accepted, you are prompted for a command:

?

Respond to the prompt with one of the PATCH commands .

NOTE

PATCH does not verify that the file you specified is a program file. A fatal error results, however, when you attempt to use a command on a non-program file.

After each command entry, PATCH prompts you for another command. To terminate PATCH, enter an invalid command or press RETURN in response to the "?" prompt.

Displays the contents of a code segment.

SYNTAX

```
?D segnum, address [,numlocations]
```

PARAMETERS

- | | |
|---------------------|--|
| <i>segnum</i> | The logical segment number to be displayed, found at the top of the PMAP output. |
| <i>address</i> | The address of the code segment you wish to display, calculated as follows: add the offset of the instruction within the procedure (an octal value found in your compiler listing), to the procedure start address found in the CODE column of your PMAP output. |
| <i>numlocations</i> | The number of words, in octal, to be displayed. |

?M

Modifies the contents of a code segment.

SYNTAX

```
?M seignum, address [,numlocations]
```

PARAMETERS

- seignum* The logical segment number to be modified, found at the top of the PMAP output.
- address* The address of the code segment you wish to modify, calculated as follows: add the offset of the instruction within the procedure (an octal value found in your compiler listing), to the procedure start address found in the CODE column of your PMAP output.
- numlocations* The number of words, in octal, to be modified.

OPERATION

In response to the M command, the contents of the current instruction are displayed, followed by a comma. To leave the value unchanged, you must re-enter the contents. To set the instruction to a "NOP" (%000000), press RETURN.

Displays the global area of the initial stack.

SYNTAX

```
?DG reloffset [numwords]
```

PARAMETERS

<i>reloffset</i>	The DB-relative offset of the word to display, found in your compiler listing.
<i>numwords</i>	The number of words, in octal, that you wish to display. The default is one.

?MG

Modifies the global area of the initial stack.

SYNTAX

```
?MG reloffset [numwords]
```

PARAMETERS

<i>reloffset</i>	The DB-relative offset of the word to modify, found in your compiler listing.
<i>numwords</i>	The number of words, in octal, that you wish to modify. The default is one.

OPERATION

In response to the MG command, the contents of the current stack word are displayed, followed by a comma. To leave the value unchanged, you must re-enter the contents. To fill the word with zeros, press **RETURN**.

EXAMPLE

In this example, all commands are used to illustrate a typical application of PATCH:

```

PROGRAM PATCH G.00.00 (C) HEWLETT-PACKARD CO., 1976
  FILE=?$OLDPASS
?DG, 0,5           Displays the values at the first five locations.
  000112
  000052
  000064
  000264
  000464

?MG, 0,5
  000112,
  000052,           Each time you press RETURN, the value is set to zero.
  000064,
  000264,
  000464,

?DG ,0,5           Displays the result of the previous ?MG command.
  000000
  000000
  000000
  000000
  000000

?MG,0,5
  000000,112       Contents changed to 112.
  000000,52
  000000,64
  000000,264
  000000,464

?DG,,1             Displays the first changed value.
  000112

END OF PROGRAM
:
```

PATCH ERROR MESSAGES

When PATCH encounters a fatal error, it will terminate with the following message:

```

ABORT :PATCH.PUB.SYS.%0.%612
PROGRAM ERROR #18 :PROCESS QUIT .PARAM = 4

PROGRAM TERMINATED IN AN ERROR STATE. (CIERR 976)
```


SLPATCH allows you to display or modify the contents of a Segmented Library (SL) file. SLPATCH requires no special user capabilities, but does run within standard MPE security restrictions. Before you use SLPATCH to alter code, refer to the MPE Segmenter Reference Manual (30000-90011).

CAUTION

SLPATCH can bypass normal MPE safeguards and modify the contents of privileged SL's. It is possible, therefore, to corrupt a file or the entire MPE operating system. HP will investigate and attempt to resolve problems resulting from modification of privileged software, but this service is not included in the standard service contract. HP will not modify the MPE operating system to accommodate problems arising from your use of PATCH on privileged files.

The System Manager must prevent unauthorized use of SLPATCH, perhaps by placing a lockword on the program file.

RUNNING SLPATCH

To use SLPATCH, you need the segment names and code displacements of your SL's. Request a PMAP of your SL by using the PREP parameter of the -ADDSL Segmenter command. The PMAP output will show you what your SL looks like, and will verify your segment names.

The segment displacement is found by adding the instruction offset (found in your compiler listing), to the starting address of the procedure, found on your PMAP listing.

1. To run SLPATCH, type:

```
:RUN SLPATCH.PUB.SYS
```

2. SLPATCH identifies itself, then prompts you with "?" for the name of the SL file you wish to modify or display. Enter the one to eight alphanumeric character SL file name, in the form *filename[.groupname[acctname]]*:

```
SLPATCH G.00.00 (C) HEWLETT-PACKARD CO., 1976  
SL FILE? SL.BOOKS.DOC
```

3. When your SL designator is accepted, you are prompted with "?" a segment name and a command. SLPATCH will prompt you for another command after each command response. To terminate SLPATCH, type EXIT in response to the "?" prompt.

?D

Displays the contents of an SL segment.

SYNTAX

```
?[segname,] D, segdisplace [,numwords]
```

PARAMETERS

<i>segname</i>	The name of the segment you want to display, obtained from an SLCREF (SL Cross-Reference) listing. This parameter must be specified the first time the segment is accessed, but may be omitted in subsequent commands.
<i>segdisplace</i>	The segment displacement, calculated as described in "RUNNING SLPATCH".
<i>numwords</i>	The number of words of the SL to display. The default is one.

Modifies the contents of an SL segment.

SYNTAX

```
?[segname,] M, segdisplace [,numwords]
```

PARAMETERS

- segname* The name of the segment you want to modify, obtained from an SLCREF (SL Cross-Reference) listing. This parameter must be specified the first time the segment is accessed, but may be omitted in subsequent commands.
- segdisplace* The segment displacement, calculated as described in "RUNNING SLPATCH".
- numwords* The number of words of the SL to modify. The default is one.

OPERATION

The contents of each word to be modified will be displayed, followed by a comma. To retain the old value, enter * and press RETURN. To enter a new value, type the number (in octal) and press RETURN. If you simply press RETURN, you will be prompted to enter "*" or a number.

SLPATCH

EXAMPLE

```
:RUN SLPATCH.PUB.SYS  
SLPATCH G.00.00 (C) HEWLETT-PACKARD CO., 1976  
SL FILE? SL.BOOKS A group-level SL is available to everyone in the group.
```

```
? SDMCOMM,D,0,4  
025001  
051404  
041605  
021040
```

```
? D,0,3 This command is also within the SDMCOMM  
025001 segment, so the segment name is not repeated.  
051404  
041605
```

```
? M,0,1 Change the contents of the first word of the segment.  
025001,025002
```

```
? EXIT
```

```
END OF PROGRAM  
:
```

SLPATCH ERROR MESSAGES

Fatal errors encountered when using SLPATCH may result in any of the following three error messages:

```
xxx IO ERROR***
```

```
***END OF FILE***
```

```
FILE ERROR=xxx
```

In these messages, "xx" is an error code returned by the FCHECK intrinsic, as described in the MPE V Intrinsic Reference Manual (32033-90007).

After the message is printed, SLPATCH terminates, and MPE prints the following:

```
ABORT :SLPATCH.PUB.SYS.%0.%1200  
PROGRAM ERROR #18 :PROCESS QUIT .PARAM = 1  
END OF PROGRAM
```

Non-fatal errors may result in one of the following messages:

```
ILLEGAL COMMAND
```

```
INVALID SL FILE
```

```
SEGMENT NOT SPECIFIED
```

```
ILLEGAL RANGE
```

```
ILLEGAL SEGMENT NAME
```

```
ILLEGAL NUMBER
```

Press **RETURN** to get the "?" prompt, then enter the correct information.

CAPABILITY SETS

Capability Sets are assigned by the System Manager. They determine the MPE functions and command set that each user may access. Your Capability Set is the result of your User Attributes, File Attributes, and Program Attributes.

User Attributes

User Attributes designate your level of access to MPE system resources.

SM	System Manager
AM	Account Manager
AL	Account Librarian
GL	Group Librarian
DI	Diagnostician
OP	System Supervisor

File Attributes

File Attributes determine your file/device interface capabilities.

SF	Permanent Files
ND	Access of non-sharable I/O devices
CV	Create volume sets
UV	Use nonsystem domain volume sets (which includes private volumes).
CS	Use communication subsystems

Program Attributes

Program Attributes determine your data structure resources (intrinsic, system tables, and instructions).

PH	Process-Handling
DS	Extra Data Segments
MR	Multiple RINs
PM	Privileged Mode
IA	Interactive Access
BA	Local Batch Access

ACCESS MODE ABBREVIATIONS

- R READ. Allows you to read files.
- L LOCK. Lets you prevent concurrent access to a file by yourself and another user. Specifically, it means that you can use the FLOCK and FUNLOCK intrinsics, and the Exclusive Access option of the FOPEN intrinsic, as described in the MPE V Intrinsics Reference Manual (32033-90007).
- A APPEND. Allows you to add data and disc extents to files, but prevents you from altering or deleting information already written. This access mode implicitly allows you to use the LOCK (L) access mode.
- W WRITE. Allows you to add, modify, or delete information on files. You can remove entire files from the system with the :PURGE command, while in this mode. The WRITE (W) access mode also implicitly gives you LOCK (L) and APPEND (A) access modes.
- S SAVE. Allows you to make existing files within your group as permanent files, with the :SAVE command or the FCLOSE intrinsic. You may rename files with the :RENAME command. You may also create new permanent files with the :BUILD command.
- E EXECUTE. Permits you to run program files, with the :RUN command or the CREATE intrinsic.

USER-TYPE ABBREVIATIONS

- ANY Any User. Any user defined in the system, including all categories defined below.
- AL Account Librarian. A user who can manage all files in the account, regardless of their groups.
- GL Group Librarian. A user who can manage certain files within his home group.
- CR Creator. The user who created this file.
- GU Group User. Any user whose home group is this group, including all GL users applicable to this group.
- AC Account Member. Any user within this account, including the AL, GU, and CR users of this account.

READER COMMENT SHEET

MPE V Utilities Reference Manual

32033-90008 July 1984

We welcome your evaluation of this manual. It is one of several that serve as a reference source for HP 3000 Computer Systems. Your comments and suggestions help us to improve our publications and will be reviewed by appropriate technical personnel. HP may make any use of the submitted suggestions and comments without obligation.

Is this manual technically accurate? Yes No (If no, explain under Comments, below.)

Are the concepts and wording easy to understand? Yes No (If no, explain under Comments, below.)

Is the format of this manual convenient in size, arrangement and readability? Yes No (If no, explain or suggest improvements under Comments, below.)

Comments:

We appreciate your comments and suggestions. This form requires no postage stamp if mailed in the U.S. For locations outside the U.S., your local HP representative will ensure that your comments are forwarded.

Date: _____

FROM:

Name _____

Company _____

Address _____

FOLD

FOLD



NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 1070 CUPERTINO, CALIFORNIA

POSTAGE WILL BE PAID BY ADDRESSEE

Documentation Manager/47U-UT
Hewlett-Packard Company
Computer Systems Division
19447 Pruneridge Avenue
Cupertino, California 95014

FOLD

FOLD

READER COMMENT SHEET

MPE V Utilities Reference Manual

32033-90008 July 1984

We welcome your evaluation of this manual. It is one of several that serve as a reference source for HP 3000 Computer Systems. Your comments and suggestions help us to improve our publications and will be reviewed by appropriate technical personnel. HP may make any use of the submitted suggestions and comments without obligation.

Is this manual technically accurate? Yes No (If no, explain under Comments, below.)

Are the concepts and wording easy to understand? Yes No (If no, explain under Comments, below.)

Is the format of this manual convenient in size, arrangement and readability? Yes No (If no, explain or suggest improvements under Comments, below.)

Comments:

We appreciate your comments and suggestions. This form requires no postage stamp if mailed in the U.S. For locations outside the U.S., your local HP representative will ensure that your comments are forwarded.

Date: _____

FROM:

Name _____

Company _____

Address _____

FOLD

FOLD



NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 1070 CUPERTINO, CALIFORNIA

POSTAGE WILL BE PAID BY ADDRESSEE

Documentation Manager/47U-UT
Hewlett-Packard Company
Computer Systems Division
19447 Pruneridge Avenue
Cupertino, California 95014



FOLD

FOLD

READER COMMENT SHEET

MPE V Utilities Reference Manual

32033-90008 July 1984

We welcome your evaluation of this manual. It is one of several that serve as a reference source for HP 3000 Computer Systems. Your comments and suggestions help us to improve our publications and will be reviewed by appropriate technical personnel. HP may make any use of the submitted suggestions and comments without obligation.

Is this manual technically accurate? Yes No (If no, explain under Comments, below.)

Are the concepts and wording easy to understand? Yes No (If no, explain under Comments, below.)

Is the format of this manual convenient in size, arrangement and readability? Yes No (If no, explain or suggest improvements under Comments, below.)

Comments:

We appreciate your comments and suggestions. This form requires no postage stamp if mailed in the U.S. For locations outside the U.S., your local HP representative will ensure that your comments are forwarded.

Date: _____

FROM:

Name _____

Company _____

Address _____

FOLD

FOLD



NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 1070 CUPERTINO, CALIFORNIA

POSTAGE WILL BE PAID BY ADDRESSEE

Documentation Manager/47U-UT
Hewlett-Packard Company
Computer Systems Division
19447 Pruneridge Avenue
Cupertino, California 95014

FOLD

FOLD

Part No. 32033-90008
Printed in U.S.A. 7/84
E0784

