RT–11 Quick Reference Manual

Order Number: AA–M241D–TC

August 1991

This manual is a quick-reference resource of summary information about the RT–11 operating system and features. Information contained in this manual has been excerpted from other volumes in the RT–11 docset.

Revision/Update Information: This manual supersedes the *RT–11 Mini-Reference Manual*, AA–M241C–TC.

Operating System: RT–11 Version 5.6

Digital Equipment Corporation
Maynard, Massachusetts

First Printing, August 1991

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation.

Digital Equipment Corporation assumes no responsibility for any errors that may appear in this document.

Any software described in this document is furnished under a license and may be used or copied only in accordance with the terms of such license. No responsibility is assumed for the use or reliability of software or equipment that is not supplied by Digital Equipment Corporation or its affiliated companies.

Restricted Rights: Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227–7013.

© Digital Equipment Corporation 1991
All rights reserved. Printed in U.S.A.

The Reader's Comments form at the end of this document requests your critical evaluation to assist in preparing future documentation.

The following are trademarks of Digital Equipment Corporation: CTS–300, DDCMP, DECNA, DECnet, DECUS, DECwriter, DEQNA, DEUNA, DIBOL, Ethernet, MASSBUS, MicroPDP–11, Micro/RSX, PDP, Professional, Q-bus, RSTS, RSX, RT–11, RTEM–11, UNIBUS, VMS, VT, and the DIGITAL logo.

S1437

This document was prepared with VAX DOCUMENT, Version 1.2.

Preface

This manual contains RT–11 system reference data excerpted from the docset. See *Associated Documents*.

Intended Audience

This information is provided for use by advanced RT–11 users, including FORTRAN IV, FORTRAN–77, and MACRO–11 assembly language programmers and C language programmers.

Document Structure

Chapter 1 — KED/KEX Editor
Major KED/KEX commands.

Chapter 2 — SL Command Line Editor
Major command-line functions: general editing, restore, movement, display.

Chapter 3 — RT–11 Commands
RT–11 keyboard monitor command summary.

Chapter 4 — RT–11 Utility Programs
Command lines, syntax, options and meanings.

Chapter 5 — Programmed Requests
EMT codes; program request descriptions, syntax, argument list definitions, error values.

Chapter 6 — Subroutines
Syntax of subroutines and functions.

Chapter 7 — Monitor Data Structures
Control blocks, disk and file formats, RMON fixed offsets, queue elements.

Chapter 8 — Standard References

ASCII character set, left/right byte equivalents; Radix–50 character set; RT–11 device names and codes; standard RT–11 file types; interrupt, trap, and software vectors; K equivalents.

Associated Documents

The RT–11 Documentation Set consists of the following associated documents:

Basic Books

- *Introduction to RT–11*
- *Guide to RT–11 Documentation*
- *RT–11 Commands Manual*
- *PDP–11 Keypad Editor User's Guide*
- *PDP–11 Keypad Editor Reference Card*
- *RT–11 Quick Reference Manual*
- *RT–11 Master Index*
- *RT–11 System Message Manual*
- *RT–11 System Release Notes*

Installation Specific Books

- *RT–11 Automatic Installation Guide*
- *RT–11 Installation Guide*
- *RT–11 System Generation Guide*

Programmer Oriented Books

- *RT–11 IND Control Files Manual*
- *RT–11 System Utilities Manual*
- *RT–11 System Macro Library Manual*
- *RT–11 System Subroutine Library Manual*
- *RT–11 System Internals Manual*
- *RT–11 Device Handlers Manual*
- *RT–11 Volume and File Formats Manual*
- *DBG–11 Symbolic Debugger User's Guide*

Conventions

The following conventions are used in this manual.

Convention	Meaning
Braces ({ })	In command syntax examples, braces enclose options that are mutually exclusive. You can choose only one option from the group of options that appears in braces.
Brackets ([ ])	Square brackets in a format line represent optional parameters, qualifiers, or values.
[/option1 /option2]	Square brackets in an option summary surrounding two or more options mean that you can specify any one of the enclosed options or any combination of them. The following example means you can specify either the /BEFORE or the /SINCE option or a combination of both to express one or two ranges of dates: [/BEFORE[:date] /SINCE[:date]]
Bold options	**Bold** options in an option summary indicate default options; that is, the option that RT–11 uses if you do not specify any choice of action.
lowercase characters	In command syntax examples, lowercase characters represent elements of a command for which you supply a value. For example: `DELETE filespec`
UPPERCASE characters	In command syntax examples, uppercase characters represent elements of a command that should be entered exactly as given.
RET	RET in examples represents the RETURN key. Unless the manual indicates otherwise, terminate all commands or command strings by pressing RET.
RETURN	RETURN in the text represents the RETURN key.
CTRL/*X*	CTRL/*X* indicates a control-key sequence. While pressing CTRL key, press another key. For example: CTRL/C

Chapter 1

KED/KEX Keypad Editor

KED is the keypad editor for an unmapped monitor; KEX is the same editor adapted to a mapped monitor.

Major KED/KEX Commands

EDIT/CREATE file	Create a file
EDIT/INSPECT file	Inspect only file
GOLD COMMAND EXIT ENTER	Save the edited file
GOLD COMMAND QUIT ENTER	Delete the edited file
GOLD COMMAND set 132 ENTER	Set screen to display 130 characters on a line.
GOLD COMMAND set 80 ENTER	Set screen to display 78 characters on a line.
GOLD COMMAND SET WRAP [number] ENTER	Set the right margin
GOLD COMMAND SET SECTION number ENTER	Set the number of lines for a section
GOLD COMMAND SET PAGE number ENTER	Set the number of lines for a page
GOLD number GOLD SPECINS	Enter the ASCII equivalent (number) of characters
GOLD BOTTOM	Move cursor to end of file
GOLD TOP	Move cursor to beginning of file
GOLD FIND character-string ADVANCE or BACKUP	Find text

Major KED/KEX Commands

GOLD COMMAND SET SEARCH EXACT ENTER	Distinguish between upper and lowercase letters
GOLD COMMAND SET SEARCH GENERAL ENTER	Not distinguish between upper and lowercase letters
GOLD COMMAND SET BOUNDED ENTER	Limit a search to a page
GOLD COMMAND SET UNBOUNDED ENTER	Extend a search to end of file
[GOLD] DELCHAR	[Un]delete a character
[GOLD] DELWORD	[Un]delete a word
[GOLD] DELLINE	[Un]delete a line
SELECT move cursor CUT	Place text in buffer
GOLD PASTE	Restore text from buffer
GOLD COMMAND CLEAR PASTE ENTER	Clear buffer
GOLD COMMAND OUTPUT filespec ENTER	Open a file for output
GOLD COMMAND CLOSE ENTER	Close a file
GOLD COMMAND WRITE SELECT ENTER	Enter text into a file opened for output
GOLD COMMAND WRITE REST ENTER	Write rest of text into a file opened for output
GOLD COMMAND WRITE number ENTER	Write number of lines into a file opened for output
GOLD COMMAND OPEN INPUT filespec ENTER	Open an input file
GOLD COMMAND INCLUDE rest ENTER	Include rest of input file in edited file
GOLD COMMAND INCLUDE number ENTER	Include number of lines from input file in edited file
LEARN [identifier]	Begin the definition of a macro
GOLD S	End the definition of a macro

GOLD M identifier	Execute the macro specified by identifier
SAVE identifier [filespec]	Copy macro to auxiliary file
LOAD identifier [filespec]	Copy auxiliary file to macro buffer
@filespec	Execute a macro directly from specified file

SL Command-Line Editor

Major Command-Line Commands

General Editing Functions

Command or Key	Function
CTRL/A	Switches between INSERT and REPLACE modes.
RECALL/ALL	Recalls all previously issued commands.
RECALL *command* or RECALL *number-in-command-stack*	Recalls a previous command.
Arrow Keys	Move the cursor through the command stack or across the command line in the direction of the arrows.

Delete Functions

Key	Function
PF4	Deletes the command line from the cursor position to the end of the command line.
CTRL/U	Deletes the command line from the cursor position to the beginning of the command line.
Keypad Comma Key [,]	Deletes one character at the cursor position.
Keypad Hyphen Key [-]	Deletes one element of command syntax at the cursor position.
LINE FEED or CTRL/J	Deletes the element of command syntax to the left of the cursor.
DELETE Key [<X]	Deletes the character to the left of the cursor.

Major Command-Line Commands

Restore Functions

Key	Function
PF1 together with the keypad key you pressed to delete information	Restores a command or command element that you deleted with a keypad function.

Movement Functions

Key	Function
CTRL/H	Switches the character at the cursor position with the character to the right of the cursor, also moving the cursor to the right.
PF1 together with ←	Moves the cursor to the beginning of the command line.
PF1 together with → or Keypad 2	Moves the cursor to the end of the preceding command line.
Keypad 1	Moves the cursor across one element of syntax.
Keypad 3	Moves the cursor one character position.
Keypad 4	Causes subsequent cursor movement to be forward.
Keypad 5	Causes subsequent cursor movement to be backward.

Display Functions

Key	Function
CTRL/E or ↑	Displays the previous command.
CTRL/R or CTRL/W	Redisplays the current command line.
PF2	Displays a help listing of SL commands.

Enable full SL (command-line) editor power with: SET SL KMON,KED,RECALL,ON

Chapter 3

RT–11 Commands

This chapter alphabetically lists RT–11 keyboard monitor commands, most of which are DCL commands.

Most command descriptions contain the following elements:

Type of Command
In parentheses underneath the command is the name of the RT–11 system component that executes that command. This is either KMON alone, a utility program(s), or a device handler.

Definition
A few sentences identifying the command.

Format
The syntax for using the command.

Option Summary
A listing of all the command's options, if it takes options.

Option Descriptions
An explanation of how to use each option with some examples.

ABORT

(KMON)

Terminates, from the system terminal, foreground and system jobs with attached private terminals.

ABORT jobname

ASSIGN

(KMON)

Defines a logical name equivalent for a physical device.

ASSIGN device-name: logical-device-name:

where:

device-name	specifies either a physical-device or a logical-device name.
logical-device-name	specifies either a logical-device name or the catch-all device assignment.

B (Base)

(KMON)

Sets a relocation base for subsequent Examine or Deposit commands.

B [address]

BACKUP

(BUP)

- Copies (backs up) and restores RT–11 files or volumes into saveset or logical-disk files.
- Initializes backup volumes when needed.
- Verifies backed-up and restored data.
- Lists directories of backup volumes and logical disks.

BACKUP in-spec out-spec

Command Options	In-Spec Options
/DEVICE /DIRECTORY [{ /OUTPUT:filespec /PRINTER }] **/FILE** /INITIALIZE /[NO]**LOG** /[NO]**QUERY** /RESTORE /[NO]**REWIND** /[NO]**SCAN** /SYSTEM /VERIFY	{ [ssname]/SAVESET[,filename,...] [ldname]/SUBSET[,filename,...] }

Table 3–1: Valid BACKUP Option Combinations

Operation	Options
BACKUP to a saveset	/[NO]SCAN /DEVICE /[NO]REWIND /VERIFY /[NO]LOG /[NO]QUERY /INITIALIZE
BACKUP to a subset	/SUBSET /VERIFY
DIRECTORY	/DIRECTORY /DIRECTORY/OUTPUT[:filespec] /DIRECTORY/PRINTER /SUBSET /SAVESET
RESTORE from a saveset	/RESTORE /SYSTEM /FILE /DEVICE /[NO]REWIND /SAVESET /VERIFY /[NO]LOG

Table 3–1 (Cont.): Valid BACKUP Option Combinations

Operation	Options
RESTORE from a subset	/SUBSET /RESTORE /SYSTEM /VERIFY /[NO]LOG
VERIFY (only)	/RESTORE/VERIFY:ONLY

/DEVICE

Backs up or restores (with the /RESTORE option) an entire volume in image mode. You can back up volumes to one or more disks, diskettes, or magtapes.

/DIRECTORY

Lists savesets on a backup volume or on a series of backup volumes.

```
.BACKUP/DIRECTORY DU1:
```

Lists files in a saveset.

```
.BACKUP/DIRECTORY DU1:TEMP.BUP/SAVESET
```

Lists files within a logical-disk file.

```
.BACKUP/DIRECTORY DU0:MYBACK.DSK/SUBSET
```

/FILE

Used only with the /RESTORE option. Restores an entire saveset to one file.

```
.BACKUP/FILE/RESTORE/VERIFY MS0:FIRST.TXT DL1:
```

/INITIALIZE

Initializes a volume for use as an output volume in a backup operation.

Use this option if your backup volume is an uninitialized backup volume or if you want to reinitialize a previously used backup volume.

```
.BACKUP/INITIALIZE DL0:F*.FOR DU1:WRK
```

/[NO]LOG

[Does not] display a list of all the files affected by a BACKUP command operation as the operation progresses.

/OUTPUT:filespec

(used only with the /DIRECTORY option) Stores a backup or logical-disk volume directory in a file.

```
.BACKUP/DIRECTORY/OUTPUT:DU0:MYBACK.DIR  DU1:MYBACK.BUP/SAVESET
```

/PRINTER

(used only with the /DIRECTORY option) Prints directory listings of backup or logical-disk volumes.

```
.BACKUP/DIRECTORY/PRINTER MU0:
```

/[NO]QUERY

[Does not] prompt you for the various responses required from the terminal.

/RESTORE

Restores backed-up data to a standard RT–11 formatted disk. You can restore:

- Complete savesets as a file or as a device

  ```
  .BACKUP/RESTORE/VERIFY MU0:28MAY.BUP/SAVESET DL1:
  ```

  ```
  .BACKUP/RESTORE DU: DU1:
  ```

- Selected files from a saveset

  ```
  .BACKUP/RESTORE/VERIFY MS0:28MAY.BUP/SAVESET,FOO.OBJ DL1:
  ```

- Selected files from a logical-disk file

  ```
  .BACKUP/RESTORE/VERIFY DL1:FRIDAY.DSK/SUBSET,MEMO1.TXT DL0:
  ```

- File-image backups not contained in a saveset or subset (see the /FILE option)

/[NO]REWIND

[Does not] rewind a magtape before performing the next operation.

/SAVESET

(used in combination with either the /RESTORE or /DIRECTORY options) Specifies the saveset containing a file(s) you want to restore or the saveset from which you want to obtain directories.

```
.BACKUP/RESTORE/VERIFY MS0:28MAY.BUP/SAVESET DL1:
```

/[NO]SCAN

[Does not] scan each output disk for bad blocks during the backup initialization procedure.

BACKUP

/SUBSET

(specified with the input file specification) Allows you to:

- Create logical-disk images of the files you want to back up

```
.BACKUP/VERIFY DU0:*.OBJ/SUBSET DU1:OBJ
```

- List directories of logical disks

```
.BACKUP/DIRECTORY DU0:MYBACK.DSK/SUBSET
```

- Restore/extract one or more files from a logical-disk file

BACKUP/RESTORE in-dev:ldname.dsk/SUBSET,file1[,file2,...] out-dev:

```
.BACKUP/RESTORE SY:WRK.DSK/SUBSET,REPORT.* DK:
```

/SYSTEM

(used only with /RESTORE) Restores SYS files when using wildcards.

/VERIFY[:ONLY]

Verifies that output data matches input data in a BACKUP/RESTORE operation.

- /VERIFY verifies a data transfer as you are backing up the data.

```
.BACKUP/VERIFY DU0:LGFIL.DAT DU1:
```

- /RESTORE/VERIFY verifies a data transfer as you are restoring the data.
- /RESTORE/VERIFY:ONLY verifies a data transfer after you back up the data but before you change or delete the original data.

BOOT

(DUP)

Loads and starts a new monitor.

BOOT input-spec

where:

input-spec specifies the device or monitor file to be bootstrapped.

Command Options

[/FOREIGN
  /WAIT]

Boot-Block Bootstrap

```
.BOOT SY: RET
RT-11XM  V05.6
```

Monitor-File Bootstrap

```
.BOOT RT11XM RET
```

/FOREIGN

Boots a pre-RT–11 V4 volume or a non-RT–11 operating system.

/WAIT

Initiates the BOOT procedure but then pauses and waits for you to mount the volume you want to bootstrap.

CLOSE

(KMON)

Closes and makes permanent all output files that are currently open in the background job.

CLOSE

The CLOSE command is particularly useful in two cases:

- After you type a CTRL/C to abort a background job
- After an unexpected program termination

In these cases, CLOSE preserves any new files that were being used by the terminated program.

The CLOSE command has no effect on a foreground job and will not make permanent any files opened on magnetic tape.

```
.R PROG RET
   .
   .
   .
CTRL/C  CTRL/C
.CLOSE  RET
```

COMPILE

(DIBOL, FORTRA, F–77, F77XM, MACRO, and CREF)

Invokes the appropriate language processor to compile the files you specify.

COMPILE filespec[s]

Command Options	Filespec Options
/LIST[:filespec] /ALLOCATE:size	/LIBRARY
/[NO]**OBJECT**[:filespec] /ALLOCATE:size	

Command Options	Filespec Options
/DIBOL	
[/ALPHABETIZE /CROSSREFERENCE /[NO]**LINENUMBERS** /LOG /ONDEBUG /PAGE:length /TABLES /[NO]**WARNINGS**]	
/FORTRAN	
[/CHECK /CODE:type /CONTINUATIONS:value /DIAGNOSE /EXTEND **/F4** or /F77 /HEADER /I4 /[NO]**LINENUMBERS** /ONDEBUG /[NO]**OPTIMIZE** /RECORD:length /SHOW[:type] /STATISTICS /[NO]**SWAP** /TRACE:type /UNITS:value /[NO]**VECTORS** /[**NO**]**WARNINGS** /WIDE /WORKFILE:value]	
/MACRO	
[/CROSSREFERENCE[:type[:type...]] /DISABLE:type[:type...] /ENABLE:type[:type...] /[NO]SHOW:type[:type...]]	

COMPILE

/ALLOCATE:size

(used with /LIST or /OBJECT) Reserves space on the device for the output file. A size of -1 reserves the largest possible space.

/ALPHABETIZE

Alphabetizes the entries in the symbol table listing.

/CHECK

(FORTRAN–77 only) Checks array references to ensure they are within the specified array address boundaries.

/CODE:type

(/FORTRAN IV only) Produces object code that is designed for a particular hardware configuration. Valid types are: EAE, EIS, FIS, and THR.

/CONTINUATIONS:value

(FORTRAN–77 only) Specifies the maximum decimal number of continuation lines in the program. The range for *value* is 0 through 99; the default is 19.

/CROSSREFERENCE[:type[:type...]]

(DIBOL or MACRO) Generates a cross reference listing of symbols; type can be:

C (Control section names)	M (Macro names)
E (Error codes)	P (Permanent symbols)
R (Register symbols)	S (User-defined symbols)

no argument — equivalent to :E:M:S

/DIAGNOSE

(FORTRAN IV only) Expands the crash dump information to include internal compiler tables and buffers.

/DIBOL

Invokes the DIBOL language processor to compile the associated files.

/DISABLE:type[:type...]

(MACRO only) Specifies .DSABL directives; type can be:

ABS	Generates absolute binary output.
AMA	Assembles absolute addresses as relative addresses.
CDR	Treats source columns beyond 72 as a comment.
DBG	Outputs internal symbol directory (ISD) records.
FPT	Truncates floating point.

GBL	Assumes undefined symbols are globals.
LC	Accepts lowercase characters in source programs.
LSB	Defines local symbol block.
MCL	Enables or disables automatic .MCALL.
PNC	Enables or disables binary output.
REG	Defines default register mnemonics.

/ENABLE:type[:type...]

(MACRO only) Specifies .ENABL directives; types are listed under /DISABLE.

/EXTEND

(FORTRAN IV or FORTRAN–77) With FORTRAN IV, extends the right margin from column 72 to column 80. With FORTRAN–77, extends the right margin of input lines from 72 to 132 columns.

/F4

Overrides any SET FORTRA conditional and calls the FORTRAN IV compiler (the default).

/F77

Overrides any SET FORTRA conditional and calls the appropriate FORTRAN–77 compiler.

/FORTRAN

Invokes the FORTRAN language processor to compile the associated files.

The FORTRAN–77 compiler under a mapped monitor is F77XM.SAV and under an unmapped monitor is F77.SAV. The FORTRAN IV compiler is FORTRA.SAV.

/HEADER

(FORTRAN IV only) Includes a list of the current options in the printout.

/I4

(FORTRAN IV or FORTRAN–77) Allocates 2-word integers.

/LIBRARY

(MACRO only) Identifies any macro library files.

/[NO]LINENUMBERS

(DIBOL or FORTRAN) [Does not] include internal sequence numbers in the executable program.

COMPILE

/LIST[:filespec]

Outputs a compiler or assembly listing to the line printer or to the filespec. (This is not the default)

/LOG

(DIBOL only) Creates a log of error messages generated by the compiler.

/MACRO

Invokes the MACRO assembler to assemble the associated files.

/[NO]OBJECT[:filespec]

[Does not] generate an OBJ file; default file name is the input file name.

/ONDEBUG

(DIBOL or FORTRAN) Includes symbol names in the DIBOL OBJ file; includes debug lines in the FORTRAN compilation.

/[NO]OPTIMIZE

(FORTRAN–77 only) [Does not] enable compiler optimization.

/PAGE:length

(DIBOL only) Defines the listing page size where length specifies the number of lines.

/RECORD:length

(FORTRAN IV or FORTRAN–77) Overrides the default record length for sequentially formatted input and output. The meaningful range for length is from 4 to 4095.

/[NO]SHOW[:type]

(FORTRAN IV, FORTRAN–77, or MACRO) Used with /FORTRAN, controls FORTRAN listing format.

FORTRAN IV Listing Options

0	Diagnostics only
1 or SRC	Source program and diagnostics
2 or MAP	Storage map and diagnostics
3	Diagnostics, source program, and storage map
4 or COD	Generated code and diagnostics
7 or ALL	All of the above

FORTRAN–77 Listing Options

0	Minimal listing; diagnostics and program section summary only
1	Source listing and program section summary
2	Source listing, program section summary, and storage map (the default)
3	Source listing, assembly code, program section summary, and section map

Used with /MACRO, to specify MACRO .LIST directives; types are as follows.

Type	Default	Controls
BEX	List	Extended binary code
BIN	List	Generated binary code
CND	List	Unsatisfied conditionals, .IF and .ENDC statements
COM	List	Comments
LD	Nolist	Listing directives with no arguments
LOC	List	Location counter
MC	List	Macro calls, repeat range expansions
MD	List	Macro definitions, repeat range expansions
ME	Nolist	Macro expansions
MEB	Nolist	Macro expansion binary code
SEQ	List	Source line sequence numbers
SRC	List	Source code
SYM	List	Symbol table
TOC	List	Table of contents
TTM	Line printer	Wide or narrow listing format

/STATISTICS

(FORTRAN IV or FORTRAN–77) Includes compilation statistics in the listing, such as amount of memory used, amount of time elapsed, and length of the symbol table.

/[NO]SWAP

(FORTRAN IV or FORTRAN–77) [Does not] permit the USR to swap over the FORTRAN program in memory.

/TABLES

(DIBOL only) Generates a symbol table and label table as part of the listing.

/TRACE:type

(FORTRAN–77 only) Enables the FORTRAN–77 /S:xxx option that controls the generation of code used by the OTS during error traceback.

/UNITS:value

(FORTRAN IV or FORTRAN–77) Overrides the default number of logical units (6) to be open at one time. With FORTRAN IV, the maximum number you can specify is 16_{10}; with FORTRAN–77, the maximum is 99_{10}.

/[NO]VECTORS

(FORTRAN IV only) [Does not] direct the FORTRAN compiler to use tables to access multidimensional arrays.

/[NO]WARNINGS

(DIBOL, FORTRAN IV, or FORTRAN–77) [Does not] include warning messages in compiler diagnostic error messages.

/WIDE

(FORTRAN–77 only) Produces a 132-column compiler listing.

/WORKFILE:value

(FORTRAN–77 only) Sets the workfile size. The default is 128_{10} blocks. By default, *value* is octal; include a decimal point after *value* to specify a decimal radix.

COPY

(PIP, DUP, and FILEX)

Transfers:

- One file to another file
- A number of files to a single file by concatenation
- Files from a large volume to several smaller volumes
- The bootstrap code on a volume to the boot blocks of that volume
- The contents of a volume to a file and vice versa

- The contents of a device to another device

COPY input-file[s] output-file

COPY Device Operations

Command Options	Input-File Options	Output-File Options
{ /BOOT[:dev] / /WAIT / /DEVICE [/FILES /WAIT] }	/END:value /START:value	/ALLOCATE:size /START:value
/IMAGE /VERIFY		

COPY Interchange Operations

Command Options	Input-File Options	Output-File Options
{ /ASCII / **/IMAGE** } /PACKED /QUERY /WAIT	{ /DOS / /OWNER:[nnn,nnn] / /INTERCHANGE / /TOPS }	{ /DOS / /INTERCHANGE[:size] }

COPY File Operations

Command Options	Input-File Options	Output-File Options
{ /ASCII /BINARY **/IMAGE** }	/POSITION:value	/POSITION:value
{ [/BEFORE[:date] /SINCE[:date]] /DATE[:date] /NEWFILES }		
/CONCATENATE		
/DELETE		
/EXCLUDE		
/IGNORE		
/INFORMATION		
/[NO]LOG		
/MULTIVOLUME		
/PREDELETE		
/[NO]PROTECTION		
/[NO]QUERY		
/[NO]**REPLACE**		
/SETDATE[:date]		
/SLOWLY		
/SYSTEM		
/VERIFY		
/WAIT		

Using Wildcards

You can use wildcards in the input or output file specification of the command. However:

- The output file specification cannot contain embedded wildcards.
- For all operations except CONCATENATE, if you use a wildcard in the input file specification, the corresponding output file name or file type must be an asterisk (*).

Specifying a Date in a COPY Option

The syntax for specifying the date is:

[dd][:mmm][:yy]

where:

dd	specifies the day (a decimal integer in the range 1–31).
mmm	specifies the first three characters of the name of the month.
yy	specifies the year (a decimal integer in the range 73–99).

/ALLOCATE:size

Reserves space for the output file; a size of -1 reserves the largest possible space.

/ASCII

Copies files in ASCII mode.

/BEFORE[:date]

Copies all files on a device created before the specified date.

```
.COPY/BEFORE:4:FEB:90 *.MAC DU0:*.MAC
```

/BINARY

Copies formatted binary files, such as OBJ files produced by the assembler.

```
.COPY/BINARY ANALYZ.OBJ DU1:*.*
```

/BOOT[:dev]

Copies bootstrap information from monitor and handler files to blocks 0 and 2 through 5 of a random-access volume, permitting you to use that volume as a system volume.

```
.COPY/BOOT DU1:RT11FB.SYS DU1:
```

/CONCATENATE

Combines several input files into a single output file.

```
.COPY/CONCATENATE DU1:*.FOR DU0:MERGE.FOR
```

/DATE[:date]

Copies only those files with the specified creation date.

```
.COPY/DATE:20:FEB:90 DU0:*.MAC DU1:*.*
```

COPY

/DELETE

Deletes the input file after it has been copied.

```
.COPY/DELETE JSPROG.SAV DU1:JSPROG.SAV
```

/DEVICE

Copies block for block the image of one device to another, and copies all the data from one disk to another without changing the file structure or the location of the files on the device.

```
.COPY/DEVICE DU0: DU1:
```

/DOS

Transfers files between RSTS/E or DOS–11 format and RT–11 format.

```
.COPY RK:PROG.BAS/DOS/OWNER:[200,200] SY:*.*
```

/END:value

Specifies the last block of the volume you are copying.

```
.COPY/DEVICE/FILES DU0:/START:0/END:500  DU1:ADAM.MAC/START:501
```

/EXCLUDE

Copies all the files on a device except the ones you specify.

```
.COPY/EXCLUDE DU0:(*.OBJ,*.SAV) DU1:*.*
```

/FILES

Copies a volume (disk image) to a file on another volume or vice versa. The following command copies the file MYWORK.BAK to DU1:

```
.COPY/DEVICE/FILES  MYWORK.BAK DU1:
```

/IGNORE

Ignores input errors during a COPY operation.

/IMAGE

Copies entire blocks.

/INFORMATION

Displays informational rather than fatal messages for input files not found and copies all others. The following command copies input files FILE1.TXT and FILE3.TXT to DU1. However, since RT–11 is unable to find DU0:FILE2.TXT, RT–11 displays a message to inform you that.

```
.COPY/INFORMATION DU0:(FILE1,FILE2,FILE3).TXT  DU1:*.*
?PIP-I-File not found DU0:FILE2.TXT
```

/INTERCHANGE[:size]

Transfers data in interchange format between interchange diskettes that are compatible with IBM 3741 format and RT–11 block-replaceable devices or the terminal. Size specifies the output record size; the default is 80 characters.

```
.COPY PROG.MAC DX1:*.*/INTERCHANGE:128
```

/[NO]LOG

[Does not] list on the terminal the names of the copied files.

```
.COPY/LOG DU1:FILE.MAC  DU0:FILE.MAC
 Files copied:
DU1:FILE.MAC  to DU0:FILE.MAC
```

/MULTIVOLUME

Copies files from an input volume to one or more output volumes. The following example shows all files on DU0 being copied to several diskettes:

```
.COPY/MULTIVOLUME DU0:*.* DU1:

(Log of files copied)

Mount next output volume in DU1:;  Continue? Y

(Log of files copied)

Mount next output volume in DU1:;  Continue? Y
```

/NEWFILES

Copies only those files that have the current system date.

```
.COPY/NEWFILES *.* DU1:*.*
 Files copied:
DK:A.FOR  to DU1:A.FOR
DK:B.FOR  to DU1:B.FOR
DK:C.FOR  to DU1:C.FOR
```

/OWNER:[nnn,nnn]

Specifies the DOS–11 user identification code (UIC) for a DOS–11 input device.

/PACKED

Copies files in DECsystem–10, DOS, or interchange mode.

/POSITION:value

Positions the magtape so that you can copy at the point you specify. The values can be 0 (tape rewinds and is searched or copied from), a positive integer (a search or copy is started at the specified point), or -1 (a search or copy is started at the current position).

/PREDELETE

Deletes files on the output device having the same names and types as those you copy to that device, before copying the input files.

/[NO]PROTECTION

[Does not] give an output file protected status so that it cannot be deleted.

/[NO]QUERY

[Does not] ask for confirmation before copying each file. The following example copies three of the four MAC files stored on DK to DU1:

```
.COPY/QUERY DK:*.MAC DU1:*.*
 Files copied:
DK:A.MAC       to DU1:A.MAC  ? Y
DK:B.MAC       to DU1:B.MAC  ? Y
DK:C.MAC       to DU1:C.MAC  ? N
DK:DEMOF1.MAC  to DU1:DEMOF1.MAC? Y
```

/[NO]REPLACE

[Does not] replace a file on the output device with the same name as the file you specify for output.

/SETDATE[:date]

Assigns the specified date to all the files copied.

/SINCE:[date]

Copies all files on a specified device that were created on or after a specified date. The following command copies only those MAC files on DK created on or after February 24, 1991:

```
.COPY/SINCE:24:FEB:91 *.MAC DU0:*.MAC
 Files copied:
DK:A.MAC  to DU0:A.MAC
DK:B.MAC  to DU0:B.MAC
DK:C.MAC  to DU0:C.MAC
```

/SLOWLY

Transfers files one block at a time.

/START[:value]

Specifies the last block of the disk you are copying. The following command copies blocks 500 to 550 of DU0 to DU1, starting at block 100:

```
.COPY/DEVICE DU0:/START:500/END:550  DU1:/START:100
```

/SYSTEM

Includes system (SYS) files when you use wildcards in an input file type, or you use the /EXCLUDE option.

```
COPY *.SYS DU1:*
```

/TOPS

Specifies DECsystem–10 DECtape input.

/VERIFY

Verifies that the output matches the input after a copy operation.

/WAIT

Waits for the specified volume to be mounted before executing the command.

CREATE

(DUP)

Creates or extends a file entry in a volume's directory.

CREATE filespec

Filespec Options

{ /EXTENSION
  /START:value
  /ALLOCATE:size }

/ALLOCATE:size

Reserves the number of blocks you specify for the file you are creating; *size* represents a decimal number of blocks. A value of -1 indicates a file of the maximum size available on the volume.

```
.CREATE DU1:myfile.type/START:117/ALLOCATE:28
```

/EXTENSION:value

Extends an existing file by the number of blocks you specify; *value* is a decimal number of blocks. At least *value* free blocks must follow the file.

Procedure for Extending a File

1. With the DIRECTORY/FULL command, determine whether there is available space adjacent to the file you want to extend:

   ```
   .DIRECTORY/FULL file-to-extend
   ```

2. With the CREATE command, extend your file:

   ```
   .CREATE DU1:myfile.type/EXTENSION:number
   ```

/START:value

Specifies the starting block number of the file you are creating. The *value* argument specifies a decimal block number. To restore a deleted file not yet overwritten, first with the DIRECTORY/DELETED command, establish the starting block numbers of the delete file:

```
.DIRECTORY/DELETED DU1:
```

D (Deposit)

(KMON)

Deposits octal values in memory, beginning at the location you specify.

D address= [value[,...value]]

The D (deposit) command accepts both word and byte addresses, but it always executes the command as though you specified a word address. (If you specify an odd address, the system decreases it by one to make it even.) The D command stores all values as word quantities. In this example, the D command deposits zeroes into locations 300, 302, 304, and 306:

```
.D 300=,,,
```

DATE

(KMON)

Sets or displays the current system date.

DATE [dd-mmm-yy])

where:

dd	represents the day (a decimal number from 1 to 31).
mmm	represents the first three characters of the name of the month.
yy	represents the year (a decimal number from 73 to 99).

The following example enters the current date:

```
.DATE 18-MAY-91
```

The next example displays the current system date:

```
.DATE
18-May-91
```

DEASSIGN

(KMON)

Removes logical name assignments made with ASSIGN. DEASSIGN with no argument removes all assignments that are currently in effect.

DEASSIGN [logical-device-name]

For example:

```
.DEASSIGN INP:
```

DELETE

(PIP, FILEX, and QUEMAN)

Deletes the files you specify from a volume's directory or the system queue.

DELETE filespec1[,filespec2,...]

RT–11 Command Options

{ [/BEFORE[:date]] }
{ [/SINCE[:date]] }
{ /DATE[:date] }
{ /NEWFILES }

/ENTRY
/EXCLUDE
/INFORMATION
/LOG
/POSITION[:sequence-number]
/[NO]QUERY
/SYSTEM
/WAIT

Interchange Command Options

{ /DOS /WAIT | /INTERCHANGE /WAIT }

/BEFORE[:date]

Deletes only those files created before the specified or (if date is omitted) the current system date. The following command deletes all SAV files on DU1 that were created before March 20, 1991:

```
.DELETE/LOG/BEFORE:20:MAR:91 DU1:*.SAV
 Files deleted:
DY0:A.SAV
DY0:B.SAV
DY0:C.SAV
```

/DOS

Deletes files from DOS–11 or RSTS/E disks or DECtapes.

/ENTRY

Deletes a job from the system queue. The following example deletes the job MILLER from the queue:

```
.DELETE/ENTRY MILLER
```

/EXCLUDE

Deletes all the files on a device except the ones you specify. The following command, for example, deletes all files from DU1 except SAV files:

```
.DELETE/EXCLUDE DU1:*.SAV
```

/INFORMATION

Displays informational rather than fatal messages for files not found and deletes all others. In the following example, the input files FILE1.TXT and FILE3.TXT are deleted. However, since RT–11 is unable to find DU1:FILE2.TXT, RT–11 displays a message to inform you:

```
.DELETE/INFORMATION DU1:(FILE1,FILE2,FILE3).TXT
?PIP-I-File not found DU1:FILE2.TXT
```

/INTERCHANGE

Deletes files from an interchange-format diskette.

DELETE

/LOG

Lists the names of all the deleted files on the terminal.

/NEWFILES

Deletes only the files with the current system date. The following example deletes the BAK files created today:

```
.DELETE/NEWFILES DU1:*.BAK
DU1:MERGE.BAK ? Y
```

/POSITION[:sequence-number]

Controls the positioning of tapes for deletions. The *sequence-number* is interpreted as the decimal sequence-number of files on a magtape.

The following table lists the operation initiated by each value of the *sequence-number* specified with the /POSITION:sequence-number option.

Sequence-Number	Operation
0	The magtape rewinds and RT–11 searches for the file you specify.
A positive integer	RT–11 starts from the magtape's present position and searches for the file you specify. If RT–11 does not find the file you specify before it reaches the *sequence-numbered* file from its starting position, it deletes the *sequence-numbered* file. **Note:** Deleting a *sequence-numbered* file also deletes any files that follow it.
A negative integer	The magtape rewinds; then RT–11 follows the procedure outlined in the preceding delete operation.

/[NO]QUERY

[Does not] ask for confirmation before executing the command. The following example shows querying. Only the file DU1:AAF.MAC is deleted:

```
.DELETE/QUERY DU1:*.*
 Files deleted:
DU1:ABC.MAC  ? N
DU1:AAF.MAC  ? Y
DU1:MERGE.FOR  ? N
```

/SINCE[:date]

Deletes only those files created on or after the specified date or, if no date is specified, the current system date. The following command deletes all SAV files on DU1 that were created on or after March 20, 1991:

```
.DELETE/LOG/SINCE:20:MAR:91 DU1:*.SAV
 Files deleted:
DU1:A.SAV
DU1:B.SAV
DU1:C.SAV
```

/SYSTEM

Allows you to delete system (SYS) files when using wildcards in an input file type. If you specify SYS as an input file type, you do not need to specify the /SYSTEM option.

/WAIT

Initiates the DELETE operation and then waits for the volume to be mounted before executing the operation.

DIBOL

(DIBOL)

Invokes the DIBOL compiler to compile one or more source programs.

DIBOL filespec[s]

Command Options

/ALPHABETIZE
/CROSSREFERENCE
/[NO]**LINENUMBERS**
  /ALLOCATE:size
/LOG
/[NO]**OBJECT**[:filespec]
  /ALLOCATE:size
/ONDEBUG
/PAGE:value
/TABLES
/[NO]**WARNINGS**

DIBOL

/ALLOCATE:size

Reserves space on the device for the output file; a size of -1 reserves the largest possible space.

/ALPHABETIZE

Alphabetizes entries in the symbol table listing.

/BUFFERING

Disables double buffering.

/CROSSREFERENCE

Generates a symbol cross-reference section in the listing. This is not the default.

/[NO]LINENUMBERS

[Does not] generate line numbers in the executable program.

/LIST[:filespec]

Outputs a program listing on the printer or in the filespec.

/LOG

Creates a log file of error messages generated by the compiler.

/[NO]OBJECT[:filespec]

[Does not] produce an OBJ file; the default file name is the input file name. In the following command, RT–11 compiles A.DBL and B.DBL together, producing files A.OBJ and B.LST. It also compiles C.DBL and produces C.LST, but does not produce C.OBJ:

```
.DIBOL A+B/LIST,C/NOOBJECT/LIST
```

/ONDEBUG

Includes a symbol table in OBJ file for use with DIBOL DDT.

/PAGE:value

Defines the listing page size; the default is 66 lines.

/TABLES

Includes label and symbol tables in the listing.

/[NO]WARNINGS

[Does not] include warning messages.

DIFFERENCES

(BINCOM and SRCCOM)

Compares two files and lists the differences between them.

DIFFERENCES 1st-filespec[s] 2nd-filespec[s]

Binary File Options

/BINARY [/ALWAYS | /BYTES | /DEVICE | /END[:value] | /QUIET | /SIPP:filespec [/ALLOCATE:size] | /START[:value]]

{ /OUTPUT:filespec [/ALLOCATE:size | /ALWAYS] | /PRINTER | **/TERMINAL** }

Text File Options

/BLANKLINES
/CASE[:option]
/CHANGEBAR
/[NO]**COMMENTS**
/FORMFEED
/MATCH[:value]

{ /OUTPUT:filespec [/ALLOCATE:size | /ALWAYS] | /PRINTER | **/TERMINAL** }

/SLP:filespec [/ALLOCATE:size | /AUDITTRAIL]

/[NO]**SPACES**
/[NO]**TRIM**

DIFFERENCES

/ALLOCATE:size

Reserves space for output file; a size of -1 reserves the largest posssible space.

/ALWAYS

Creates an output file whether or not differences exist.

/AUDITTRAIL

Marks any changes made by SLP so that you have a record of how you patch source file.

/BINARY

Compares two binary files and lists the differences between them.

/BLANKLINES

Includes blank lines in the file comparison.

/BYTES

Lists the differences byte by byte.

/CASE[:option]

Controls case sensitivity in a file comparison. The two options are EXA for case-sensitive comparisons and GEN for case-insensitive comparisons.

/CHANGEBAR

Creates an output file that marks additions and deletions in the second file. A vertical bar marks an addition; a bullet (lowercase letter *o*) marks a deletion. The following example command creates a listing of RTLIB.MAC with a changebar or bullet character at the left margin of each line that is different from RTLIB.BAK:

```
.DIFFERENCES/CHANGEBAR RTLIB.BAK RTLIB.MAC
```

/[NO]COMMENTS

[Does not] include in the file comparison all assembly-language comments found in the two files.

/DEVICE

Compares two entire volumes starting with block 0.

/END[:value]

Specifies the last block to compare.

/FORMFEED

Includes form feeds in the output listing.

/MATCH[:value]

Specifies the number of lines from each file that must agree to constitute a match. The default is 3.

/OUTPUT:filespec

Specifies a device and file name for the differences listing. The default file type is DIF. The following command compares two text files and creates a file EXAMP.DIF containing a listing of any differences, if there are differences:

```
.DIFFERENCES/OUTPUT:EXAMP EXAMP.ONE EXAMP.TWO
```

/PRINTER

Prints a listing of the differences on the printer.

/QUIET

Suppresses the display of the differences at the terminal.

/SIPP:filespec

Creates a command file for SIPP, the save-image patch program. The following command creates a command file which, when run with SIPP, patches DEMOF1.BAK so it matches DEMOF1.SAV:

```
DIFFERENCES/BINARY/SIPP:PATCH.COM  DEMOF1.BAK DEMOF1.SAV
```

/SLP[:filespec]

Creates a command file for SLP, the source-language patch program (SLP). The following command creates the command file PATCH.SLP, which can be used as input to the SLP program to patch RTLIB.BAK so that it matches RTLIB.MAC:

```
.DIFFERENCES/SLP:PATCH RTLIB.BAK RTLIB.MAC
```

/[NO]SPACES

[Does not] include spaces and tabs in the file comparison.

/START[:value]

Specifies the octal starting block number of the file comparison.

/TERMINAL

Displays the list of differences on the terminal.

/[NO]TRIM

[Does not] ignore tabs and spaces at the ends of lines.

DIRECTORY

(DIR)

Lists volume directory information.

DIRECTORY filespec1, filespec2, ...

RT–11 Directory Options

/BEGIN

/BADBLOCKS

[/END:value
/FILES
/START:value
/WAIT]

{ /ALPHABETIZE
  /REVERSE
/ORDER[:category]
  /REVERSE
/POSITION
/SORT[:category]
  /REVERSE }

{ [/BEFORE[:date]
  /SINCE[:date]]
/DATE[:date]
/NEWFILES }

/BLOCKS
/BRIEF
/COLUMNS
/DELETED
/EXCLUDE
/FAST
/FREE
/FULL
/OCTAL
/[NO]PROTECTION
/SUMMARY
/VOLUMEID[:ONLY]

RT–11 Directory Options

{ /OUTPUT:filespec
  /ALLOCATE:size
  /PRINTER
  /TERMINAL }

Command Options for Interchange Directories

/BRIEF
{ /DOS
  /OWNER:[nnn,nnn]
  /WAIT
  /INTERCHANGE
  /WAIT
  /TOPS
  /WAIT }

/FAST
{ /OUTPUT:filespec
  /ALLOCATE:size
  /PRINTER
  /TERMINAL }

/ALLOCATE:size

Reserves space for the output listing file; a size of -1 reserves the largest possible space.

/ALPHABETIZE

Lists directory entries in alphabetical order.

/BADBLOCKS

Scans a volume for badblock and lists their block numbers.

/BEFORE[:date]

Lists files created before date. The following command lists on the terminal all files stored on device DU1 created before February 1991:

```
.DIRECTORY/BEFORE:1:FEB:91 DU1:
```

DIRECTORY

/BEGIN

Lists the directory beginning at the file you specify. The following example lists the file SRCCOM.SAV on device DU1 and all the files that follow it in the directory:

```
.DIRECTORY DU1:SRCCOM.SAV/BEGIN
```

/BLOCKS

Includes the starting block numbers of files and free areas in the directory listing.

/BRIEF

Lists only file names and types; equivalent of /FAST.

/COLUMNS:value

Lists a directory in the specified number of columns (from 1 to 9).

```
.DIRECTORY/COLUMNS:1 DU1:
```

/DATE[:date]

Lists only those files created on the specified date. The default is the current date. The following command lists all the files on device DU1 that were created on February 15, 1991:

```
.DIRECTORY/DATE:15:FEB:91 DU1:
```

/DELETED

Lists files that have been deleted:

```
.DIRECTORY/DELETED DU1:
```

/DOS

Lists the directory of a device that is in RSTS/E or DOS–11 format.

/END:value

Specifies the final block of a bad-block scan.

/EXCLUDE

Lists all files except those you specify. The following command lists all files on DU1 except the SAV and SYS files:

```
.DIRECTORY/EXCLUDE DU1:(*.SAV,*.SYS)
```

/FAST

Lists only file names and types; equivalent of /BRIEF.

/FILES

Lists names of files having bad blocks:

```
.DIRECTORY/BADBLOCKS/FILES DU1:
```

/FREE

Lists a directory of unused areas and the size of each.

/FULL

Lists file names, free areas, sizes, and creation dates. The following example lists the entire directory for device DU1:

```
.DIRECTORY/FULL DU1:
 15-Feb-91
RT11XM.SYS   123P 31-Oct-90      SWAP  .SYS    28P 31-Oct-90
DU    .SYS    10P 31-Oct-90      SL    .SYS    17P 31-Oct-90
LD    .SYS    11P 31-Oct-90      PIP   .SAV    30P 31-Oct-90
DUP   .SAV    52P 31-Oct-90      DIR   .SAV    19P 31-Oct-90
RESORC.SAV    32P 31-Oct-90      MEMO1 .TXT     2  15-Feb-91
< UNUSED >     2                 MEMO3 .TXT     2  15-Feb-91
< UNUSED >   458
 11 Files, 326 Blocks
 460 Free blocks
```

/INTERCHANGE

Lists the directory of a diskette that is in interchange format.

/NEWFILES

Lists only those files with the current system date:

```
.DIRECTORY/NEWFILES DU1:
```

/OCTAL

Lists in octal sizes and, if /BLOCK is used, starting blocks.

/ORDER[:category]

Sorts the directory listing by category; category can be any of the following.

Category	Function
DATE	creation date
NAME	file name
POSITION	file position on the volume
SIZE	file size in blocks
TYPE	file type

The following example lists the directory of device DU1, according to date:

```
.DIRECTORY/ORDER:DATE DU1:
```

/OUTPUT:filespec

Specifies a device and file name for the listing file. Normally, the directory listing appears on the terminal.

/OWNER:[nnn,nnn]

Specifies a user identification code (UIC) for a DOS formatted volume; the square brackets are a part of the UIC.

/POSITION

Lists the files in the order that they occur on the volume (the default); includes file sequence numbers of files stored on a magtape or starting block numbers of files on disk.

/PRINTER

Prints the directory listing on the printer.

/[NO]PROTECTION

Lists all [un]protected files on a volume.

```
.DIRECTORY/ORDER:SIZE/REVERSE/PROTECTION
 15-Feb-91
RT11XM.SYS   123P 31-Oct-90      DIR   .SAV    19P 31-Oct-90
DUP   .SAV    52P 31-Oct-90      SL    .SYS    17P 31-Oct-90
RESORC.SAV    32P 31-Oct-90      LD    .SYS    11P 31-Oct-90
PIP   .SAV    30P 31-Oct-90      DU    .SYS    10P 31-Oct-90
SWAP  .SYS    28P 31-Oct-90
 9 Files, 322 Blocks
 460 Free blocks
```

/REVERSE

Lists a directory in the reverse order of the sort you specify with /ALPHABETIZE, /ORDER, or /SORT.

/SINCE[:date]

Lists all files created on or after the specified date.

```
.DIRECTORY/SINCE:13:AUG:90
```

/SORT[:category]

Equivalent to /ORDER.

/START:value

Specifies the starting block for a bad-block scan.

/SUMMARY

Lists the number of files, the blocks in use, and the free blocks on a volume.

```
.DIRECTORY/SUMMARY
 14-Mar-91

44 Files in segment 1
46 Files in segment 4
37 Files in segment 2
34 Files in segment 5
38 Files in segment 3
16 Available segments, 5 in use

199 Files, 3647 Blocks

1115 Free blocks
```

/TOPS

Lists the directory of a DECsystem–10 DECtape.

/VOLUMEID[:ONLY]

Includes the volume ID and owner name at the beginning of the directory listing; with ONLY, lists just the volume ID and the owner name.

```
.DIRECTORY/VOLUMEID DU1:
 15-Feb-91
 Volume ID: RT11A
 Owner    : Harry
RT11XM.SYS   123P 31-Oct-90      SWAP   .SYS    28P 31-Oct-90
DU    .SYS    10P 31-Oct-90      SL     .SYS    17P 31-Oct-90
LD    .SYS    11P 31-Oct-90      PIP    .SAV    30P 31-Oct-90
DUP   .SAV    52P 31-Oct-90      DIR    .SAV    19P 31-Oct-90
RESORC.SAV    32P 31-Oct-90      MEMO1  .TXT     2  15-Feb-91
MEMO3 .TXT     2  15-Feb-91
 11 Files, 326 Blocks
 460 Free blocks
```

/WAIT

Waits for the volume to be mounted before executing the command.

DISMOUNT

(KMON)

Frees a logical-disk unit number from its associated file.

DISMOUNT logical-disk-unit

For example:

```
.DISMOUNT LD3:
```

or

```
.ASSIGN LD3: DAT
.DISMOUNT DAT
```

DUMP

(DUMP)

Lists the contents of file in octal words, octal bytes, ASCII characters, or Radix–50 characters.

DUMP filespec

Command Options

/[NO]**ASCII**
/BYTES
/END:value
/FOREIGN
/IGNORE
/ONLY:value
{ /OUTPUT:filespec [/ALLOCATE:size] | **/PRINTER** | /TERMINAL }
/RAD50
/START:value
/WORDS

/ALLOCATE:size
Reserves space for the output file; a size of -1 reserves the largest possible space.

/[NO]ASCII
[Does not] list the ASCII equivalent of each octal word or byte.

/BYTES
Lists information in octal bytes.

/END:value
Specifies the last block to dump.

/FOREIGN
Dumps a magtape that is not RT–11 file-structured.

/IGNORE
Ignores I/O errors that occur during a dump operation.

/ONLY:value
Dumps only the block you specify.

/OUTPUT:filespec
Sends the output to the fiilespec.

/PRINTER
Sends the output to the printer.

/RAD50
Lists the Radix–50 equivalent of each octal word.

/START:value
Specifies the first block to dump.

/TERMINAL
Sends the output to the terminal.

/WORDS
Lists information in octal words.

E (Examine)

(KMON)

Displays in octal the contents of an address on the console terminal.

E address1[-address2])

For example:

```
.E 1000
127401
```

or

```
.E 1001-1005
127401 007624 127400
```

EDIT

(EDIT, KED, KEX, TECO)

Runs a text editor for creating or modifying ASCII files. KED is the default editor for an unmapped monitor; KEX is the default editor for a mapped monitor.

EDIT filespec

Command Options	Filespec Options
{ /EDIT { /KED } [/[NO]**COMMAND**[:init-filespec] { /KEX } [/JOURNAL[:jour-filespec] [/[NO]**QUERY** [/RECOVER[:jour-filespec] /TECO /EXECUTE:filespec }	/ALLOCATE:size
{ /CREATE /INSPECT (or /READONLY) /OUTPUT:filespec /ALLOCATE:size }	

/ALLOCATE:size

Reserves space for the output file; a size of -1 reserves the largest possible space.

```
.EDIT/CREATE DU1:PROG1.MAC/ALLOCATE:-1
```

/[NO]COMMAND[:init-filespec]

(KED/KEX only) [Does not] execute the initialization file KEDINI.KED or the specified initialization file. The following command executes the initialization file *myinit.ked* before beginning the work session:

```
EDIT/COMMAND:myinit.ked myfile.txt
```

/CREATE

Creates a new file when the specified input file is not found.

```
.EDIT/CREATE NEWFIL.TXT
```

/EDIT

Invokes the EDIT line editor.

/EXECUTE:filespec

Executes TECO commands contained in the file you specify.

/INSPECT

Opens a file with read-only access.

```
.EDIT/INSPECT NEWFIL.TXT
```

/JOURNAL[:jour-filespec]

(KED/KEX only) Produces a journal file that will let you recover your work after a system crash or similar malfunction. The default journal file name is your output file name with a JOU file type.

```
.EDIT/JOURNAL PROG1.FOR
```

/KED

Invokes the KED keypad editor, the default editor for unmapped monitors:

```
.EDIT/KED PROG1.FOR
```

/KEX

Invokes the KEX keypad editor, the default editor for mapped monitors:

```
.EDIT/KEX PROG1.FOR
```

EDIT

/OUTPUT:filespec

Specifies a new file to contain the output of your work session, leaving the input file unchanged:

```
.EDIT/OUTPUT:PROG2.FOR PROG1.FOR
```

/[NO]QUERY

[Does not] request confirmation before executing some commands, such as creating a new file:

```
.EDIT/NOQUERY MEMO20.TXT
```

/READONLY

Opens a file as read only. This is the same as /INSPECT.

/RECOVER[:jour-filespec]

Produces a text file containing the work in your editing session that was terminated without an EXIT or QUIT command. In executing the following command, RT–11 looks for the journal file MEMO20.JOU:

```
.EDIT/RECOVER MEMO20.TXT
```

/TECO

Invokes the TECO editor. (TECO is not supported or distributed by Digital.)

EXECUTE

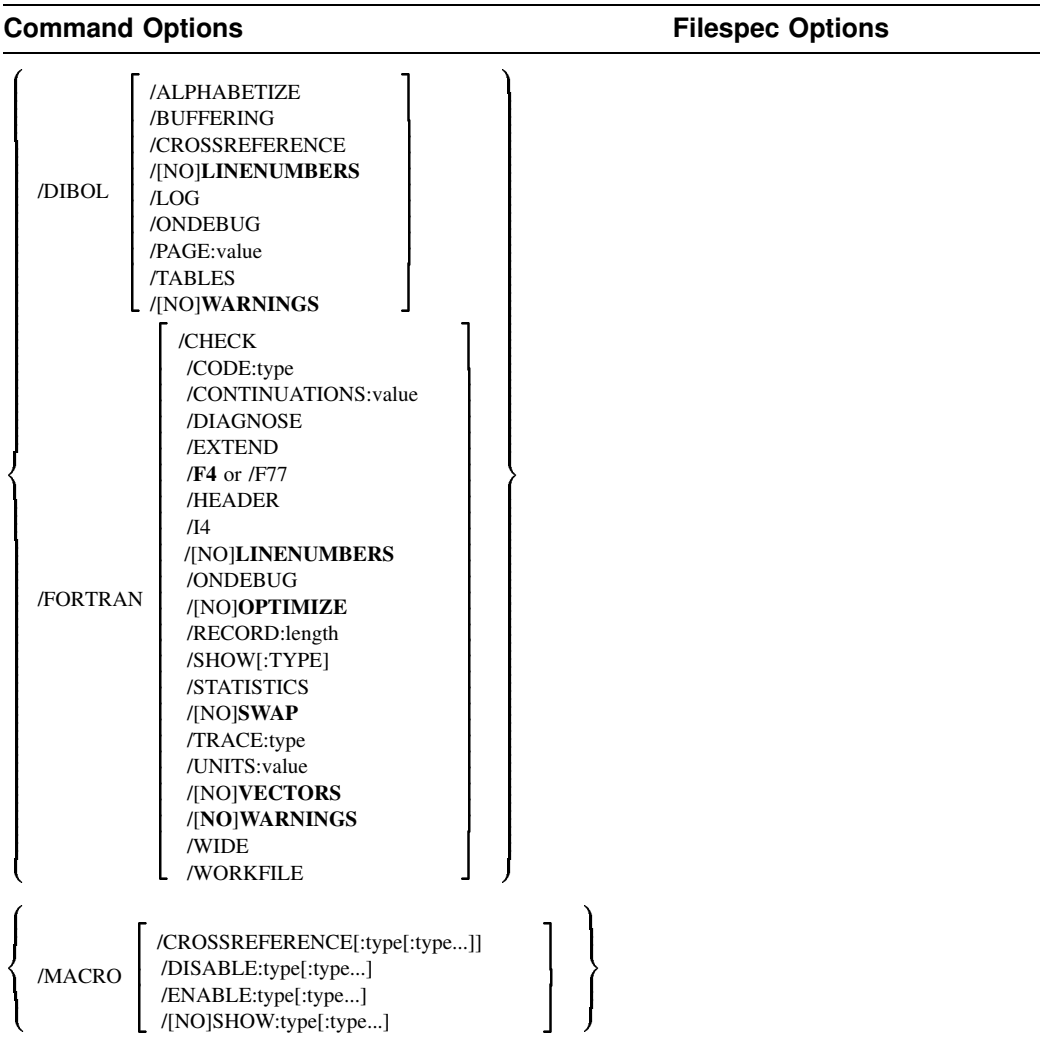
(DIBOL, CREF, F77, F77XM, FORTRA, LINK, and MACRO)

Invokes one or more language processors, compiles or assembles source files, and then links and runs the resulting modules.

EXECUTE filespec[s]

Command Options	Filespec Options
/BOTTOM:value	[/LIBRARY[:filespec]]
/DEBUG[:filespec]	
/DUPLICATE	
/EXECUTE[:filespec]	
/ALLOCATE:size	
/GLOBAL	
/LINKLIBRARY:filespec	
/LIST[:filespec]	
/ALLOCATE:size	
/MAP[:filespec]	
/ALLOCATE:size	
/WIDE	
/OBJECT[:filespec]	
/ALLOCATE:size	
/PROMPT	
/[NO]**RUN**	

EXECUTE



/ALLOCATE:size

Reserves space for the output file; a size of -1 reserves the largest possible space.

/ALPHABETIZE

(DIBOL only) Alphabetizes the symbol table listing.

/BOTTOM:value

Specifies the lowest address to be used by the relocatable code in the load module.

/BUFFERING

(DIBOL only) Disables double buffering.

/CHECK

(FORTRAN–77 only) Checks array references to ensure they are within the specified array address boundaries.

/CODE:type

(FORTRAN IV only) Controls type of code generated; type can be EAE, EIS, and THR.

/CONTINUATIONS:value

(FORTRAN–77 only) Specifies the maximum decimal number of continuation lines in the program. The range is 0 through 99; the default is 19.

/CROSSREFERENCE[:type[:type...]]

(DIBOL and MACRO) Generates a cross reference listing of symbols; type can be:

C (Control section names)	M (Macro names)
E (Error codes)	P (Permanent symbols)
R (Register symbols)	S (User-defined symbols)

no argument — equivalent to :E:M:S

/DEBUG[:filespec]

Links ODT [or another debugger] with your program.

/DIAGNOSE

(/FORTRAN IV only) Expands the crash dump information to include internal compiler tables and buffers.

/DIBOL

Runs the DIBOL compiler.

/DISABLE:type[:type...]

(MACRO only) Specifies .DSABL directives; type can be:

ABS	Generates absolute binary output.
AMA	Assembles absolute addresses as relative addresses.
CDR	Treats source columns beyond 72 as a comment.
DBG	Outputs internal symbol directory (ISD) records.
FPT	Truncates floating point.

GBL	Assumes undefined symbols are globals.
LC	Accepts lowercase characters in source programs.
LSB	Defines local symbol block.
MCL	Enables or disables automatic .MCALL.
PNC	Enables or disables binary output.
REG	Defines default register mnemonics.

/DUPLICATE

Allows multiple copies of library modules when linking.

/ENABLE:type[:type...]

(MACRO only) Specifies .ENABL directive; types are listed under /DISABLE.

/EXECUTE[:filespec]

Specifies a file or device for the executable file; the default file name is the input file name. The following command creates an executable file called PROG1.SAV on device DU1:

```
.EXECUTE/EXECUTE:DU1:  PROG1,PROG2
```

The next command creates an executable file called MYPROG.SAV on device DK:

```
.EXECUTE RTN1,RTN2,MYPROG/EXECUTE
```

/EXTEND

(/FORTRAN IV) Extends the right margin of input lines from column 72 to column 80.

(/FORTRAN–77) Extends the right margin of input lines from column 72 to column 132.

/F4

Overrides any SET FORTRA conditional and calls the FORTRAN IV compiler (the default).

/F77

Overrides any SET FORTRA conditional and calls the appropriate FORTRAN–77 compiler.

/FORTRAN

Runs the FORTRAN compiler.

/GLOBAL

Includes a cross reference of global symbols as a part of the load map listing. The following command produces a map listing file, MYPROG.MAP, that contains a global symbol cross-reference section:

```
.EXECUTE/GLOBAL/MAP:DU1:  MYPROG
```

/HEADER

(/FORTRAN IV only) Includes a list of the current options in the printout.

/I4

(FORTRAN IV or FORTRAN–77) Allocates 2-word integers.

/LIBRARY

(MACRO only) Identifies the file this option qualifies as a macro library file:

```
.EXECUTE A+B/LIST/OBJECT,MYLIB/LIBRARY+C.MAC/LIST/OBJECT
```

/[NO]LINENUMBERS

(DIBOL, FORTRAN IV, or FORTRAN–77) [Does not] include line numbers in the executable program.

/LINKLIBRARY:filespec

Includes the file as a library at link time.

/LIST[:filespec]

Produces a compilation or assembly listing on the printer or to the filespec. The following command creates a listing file called A.LST on DU1:

```
.EXECUTE/LIST:DU1: A.MAC
```

/LOG

(DIBOL only) Creates a log file of error messages generated by the compiler.

/MACRO

Runs the MACRO assembler.

/MAP[:filespec]

Sends a load map to the printer or to the filespec.

/OBJECT[:filespec]

Specifies a file name for the OBJ file.

/ONDEBUG

(DIBOL, FORTRAN IV, or FORTRAN–77) Includes symbol names in the output file for use with DIBOL DDT; includes debug lines in FORTRAN compilation.

EXECUTE

/[NO]OPTIMIZE
(FORTRAN–77 only) [Does not] enable compiler optimization.

/PAGE:value
(DIBOL only) Specifes the number of lines on a listing page; the default is 66.

/PROMPT
Accepts input lines until a terminating double slash (//) is found.

/RECORD:length
(FORTRAN IV or FORTRAN–77) Overrides the default record length of 132 characters; valid range is from 4 to 4095.

***/[NO]*RUN**
[Does not] run the program after linking.

/[NO]SHOW:type
(MACRO only) Specifies the MACRO [.NLIST] and .LIST directives; type can be:

BEX	Extended binary code
BIN	Generated binary code
CND	Unsatisfied conditions and .IF and .ENDC statements
COM	Comments
LOC	Location counter
LD	Listing directives with no arguments
MC	Macro calls, repeat range expansions
MD	Macro definitions, repeat range expansions
ME	Macro expansions
MEB	Macro expansions, binary code
SEQ	Source line sequence numbers
SRC	Source code
SYM	Symbol table
TOC	Table of contents
TTM	Wide or narrow listing format

/SHOW:type
(FORTRAN IV, FORTRAN–77) Controls FORTRAN listing format. See the FORTRAN command description for the types you can use.

/STATISTICS
(FORTRAN IV or FORTRAN–77) Includes compilation statistics in the listing.

/[NO]SWAP

(FORTRAN IV or FORTRAN–77) [Does not] permit the USR to swap over the FORTRAN program in memory.

/TABLES

(DIBOL only) Includes lable and symbol tables in the listing.

/TRACE:type

(FORTRAN–77 only) Enables the FORTRAN–77 /S:xxx option.

/UNITS:value

(FORTRAN IV or FORTRAN–77) Specifies the number of FORTRAN logical units; the initial value is 6; maximum is 16 (for FORTRAN IV) or 99 (for FORTRAN–77).

/[NO]VECTORS

(FORTRAN IV only) [Does not] access FORTRAN multidimensional arrays by building tables of pointers.

/[NO]WARNINGS

(DIBOL, FORTRAN IV, or FORTRAN–77) [Does not] include warning messages.

/WIDE

(FORTRAN–77 only) Produces a 132-column compiler listing.

/WORKFILE:value

(FORTRAN–77 only) Sets the workfile size; the default is 128_{10} blocks. Include a decimal point after *value* to specify a decimal radix.

FORMAT

(FORMAT)

- Formats some disks and diskettes by writing headers to each block in the volume
- Verifies that blocks on disks and diskettes can be read from or written to.
- Replaces bad blocks (ones from/to which you cannot read/write) on a device with good blocks.

FORMAT device:

FORMAT

Command Options

/[NO]**QUERY**
/SINGLEDENSITY
/VERIFY[:ONLY]
  /PATTERN:value
/WAIT

You can FORMAT and /VERIFY the following devices:

DW devices (Devices for Professional 300 Series processors)
RK05
RK06/RK07
RX01 and RX02 on RX02 drives
RX33

You can *only* /VERIFY [:ONLY] the following devices:

RL01/RL02
RX01
RX50

/PATTERN[:value]

specifies which of the following 16-bit patterns to use for verification:

Pattern	Bit Set	Value	16-Bit Pattern
1	0	1	000000
2	1	2	177777
3	2	4	163126
4	3	10	125252
5	4	20	052525
6	5	40	007417
7	6	100	021042
8	7	200	104210
9	8	400	155555
10	9	1000	145454
11	10	2000	146314
12	11	4000	162745
13	12	10000	*
14	13	20000	*
15	14	40000	*
16	15	100000	*

*These patterns are reserved for future use. Currently these bit patterns run the default bit pattern (pattern 8).

```
.FORMAT/VERIFY/PATTERN:25  DL0:
DL0:/FORMAT-Are you sure? Y
?FORMAT-I-Formatting complete
PATTERN  5
PATTERN  3
PATTERN  1
?FORMAT-I-Verification complete
```

/[NO]QUERY

[Does not] ask for confirmation before formatting.

FORMAT

/SINGLEDENSITY

Formats diskette in single-density format:

```
.FORMAT/SINGLEDENSITY DY1:
```

/VERIFY[:ONLY]

Verifies after formatting; with ONLY, verifies without formatting; but its verification procedure is destructive of information already on the volume.

```
.FORMAT/VERIFY DU1:
DU2:/FORMAT-Are you sure? Y
?FORMAT-I-Formatting complete
PATTERN  8
?FORMAT-I-Verification complete
```

/WAIT

Waits for the volume to be mounted before executing the command.

```
.FORMAT/WAIT DU1:
DU1:/FORMAT-Are you sure? Y
Mount input volume in <device>; Continue? Y
?FORMAT-I-Formatting complete
Mount system volume in <device>; Continue? Y
```

FORTRAN

(FORTRA.SAV, F77.SAV, F77XM.SAV)

Runs a FORTRAN (FORTRAN IV or FORTRAN–77) compiler (FORTRAN IV by default).

FORTRAN filespec[s]

FORTRAN IV Command Options	FORTRAN–77 Command Options
/CODE:type	/CHECK
/DIAGNOSE	/CONTINUATIONS
/EXTEND	/EXTEND
/F4	/F77
/HEADER	/I4
/I4	/[NO]**LINENUMBERS**
/[NO]**LINENUMBERS**	/LIST[:filespec]
/LIST[:filespec]	/ALLOCATE:size
/ALLOCATE:size	/[NO]OBJECT[:filespec]
/[NO]OBJECT[:filespec]	/ALLOCATE:size
/ALLOCATE:size	/ONDEBUG
/ONDEBUG	/[NO]**OPTIMIZE**
/RECORD:length	/RECORD:length
/SHOW[:type]	/SHOW[:type]
/STATISTICS	/STATISTICS
/[NO]**SWAP**	/[NO]**SWAP**
/UNITS:value	/TRACE:type
/[NO]**VECTORS**	/UNITS:value
/[**NO**]**WARNINGS**	/[**NO**]**WARNINGS**
	/WIDE
	/WORKFILE:value

/ALLOCATE:size

Reserves space for the output file; a size of -1 reserves the largest possible space.

/CHECK

(FORTRAN–77 only) Checks array references to ensure they are within the specified array address boundaries.

/CODE:type

(FORTRAN IV only) Control type of code generated; type can be EAE, EIS, FIS, or THR.

FORTRAN

/CONTINUATIONS:value

(FORTRAN–77 only) Specifies the maximum decimal number of continuation lines in the program. The range is 0 through 99_{10}; the default is 19.

/DIAGNOSE

(FORTRAN IV only) Expands the crash dump information.

/EXTEND

Extends the right margin of input lines from 72 to 80 columns (for FORTRAN IV) and from 72 to 132 columns (for FORTRAN–77).

/F4

Calls the FORTRAN IV compiler (the default).

/F77

Calls the appropriate FORTRAN–77 compiler.

/HEADER

(FORTRAN IV only) Includes a list of the current options in the printout.

/I4

Allocates 2-word integers.

/[NO]LINENUMBERS

[Does not] include line numbers in the executable program.

/LIST[:filespec]

Sends the compiler listing to the printer or to the file specification.

/[NO]OBJECT[:filespec]

[Does not] generate an OBJ file.

/ONDEBUG

Includes debug lines (those that have a D in column one) in the compilation.

/[NO]OPTIMIZE

(FORTRAN–77 only) [Does not] enable compiler optimization.

/RECORD:length

Overrides the default record length of 132 characters; the valid range is from 4 to 4095.

/SHOW[:type]

Controls FORTRAN listing format; the type codes are as follows.

FORTRAN IV Listing Options	
0	Diagnostics only
1 or SRC	Source program and diagnostics
2 or MAP	Storage map and diagnostics
3	Diagnostics, source program, and storage map
4 or COD	Generated code and diagnostics
7 or ALL	All of the above

FORTRAN–77 Listing Options	
0	Minimal listing; diagnostics and program section summary only
1	Source listing and program section summary
2	Source listing, program section summary, and storage map (the default)
3	Source listing, assembly code, program section summary, and section map

/STATISTICS

Includes compilation statistics in the listing.

/[NO]SWAP

[Does not] permit the USR to swap over the FORTRAN program in memory.

/TRACE:type

(FORTRAN–77 only) Enables the FORTRAN–77 /S:xxx option; types can be:

Type Value	Description of Compiled Traceback Information
(omitted)	Same as the /NOLINENUMBERS option
ALL	All source statements, functions, and subroutine entries
LIN	Same as ALL
BLO	Selected source statements, all functions, and all subroutine entries (the default)
NAM	All function and subroutine entries
NON	None compiled

FORTRAN

/UNITS:value

Specifies the number of FORTRAN logical units; the initial value is 6; maximum is 16 (for FORTRAN IV) or 99 (for FORTRAN–77).

/[NO]VECTORS

(FORTRAN IV only) [Does not] access FORTRAN multidimensional arrays by building tables of pointers.

/[NO]WARNINGS

[Does not] include warning messages.

/WIDE

(FORTRAN–77 only) Produces a 132-column compiler listing.

/WORKFILE:value

(FORTRAN–77 only) Sets the workfile size; the default is 128_{10} blocks. Include a decimal point after *value* to specify a decimal radix.

FRUN

(KMON)

Initiates foreground jobs on multi-job monitors. See also R, RUN, SRUN, V/VRUN.

FRUN filespec

Filespec Options

/BUFFER:value
/NAME:name
/PAUSE
/TERMINAL:value

/BUFFER:value$_8$
Reserves more space (words) in memory than the actual program size.

/NAME:name
Assigns a logical name to the foreground job.

/PAUSE
Displays the load address of your foreground job and waits for a RESUME.

```
.FRUN DEMOSP/PAUSE
Loaded at 127276
.RESUME
```

/TERMINAL:value
Assigns the specified terminal to the foreground job.

GET

(KMON)

Loads a memory image file into memory.

GET filespec

HELP

(HELP)

Displays concise DCL command descriptions with examples.

HELP [topic [subtopic[:item[:item...]]]]

Command Options

/TERMINAL
/PRINTER

/PRINTER

Prints the help information on the printer.

/TERMINAL

Displays the HELP information on the terminal.

1. The HELP * command lists and briefly describes all the RT–11 DCL commands:

```
.HELP *

ABORT        Terminates, from the system console, a Foreground job
ASSIGN       Associates a logical device name with a physical device
B            Sets a relocation base
BACKUP       Backup/Restore large files on random access devices
  .
  .
  .
```

2. The HELP DATE command displays a brief description of the DATE command:

```
.HELP DATE

DATE          Sets or displays the current system date

  SYNTAX
       DATE[ dd-mmm-yy]

  SEMANTICS
       All numeric values are decimal; mmm represents the
       first three characters of the name of the month.  Under
       RTEM-11, the current date cannot be changed.

  OPTIONS
       None

  EXAMPLES
       DATE 12-MAR-90
```

INITIALIZE

(DUP and FILEX)

Writes an RT–11 empty device directory on the specified volume.

INITIALIZE device:

RT–11 Command Options

/BADBLOCKS[:RETAIN]
/FILE:filespec
/[NO]**QUERY**
/REPLACE[:RETAIN]
/SEGMENTS:value
/VOLUMEID[:ONLY]
/WAIT
/RESTORE

Interchange Command Options

{ /DOS | /WAIT | /INTERCHANGE | /WAIT }
/[NO]**QUERY**
/VOLUMEID[:ONLY]

/BADBLOCKS[:RETAIN]

Scans a volume for bad blocks and creates FILE.BAD directory entries; with RETAIN, retains the existing FILE.BAD files but does not scan for bad blocks. If the volume being initialized contains bad blocks, RT–11 displays the locations of the bad blocks in octal and in decimal, as in the following example:

```
.INITIALIZE/BADBLOCKS DL0:
DL0:/Initialize;  Are you sure?  Y
    Block         Type
000120     80.  Hard
000471    313.  Hard
000521    337.  Hard
?DUP-W-Bad blocks detected 3.
```

/DOS

Initializes a DECtape for DOS–11 format.

INITIALIZE

/FILE:filespec

Initializes a magtape and creates a bootable tape. The following example creates a bootable magtape on an MS device:

```
.INITIALIZE/FILE:MSBOOT.BOT  MS0:
```

/INTERCHANGE

Initializes a diskette for interchange format.

/[NO]QUERY

[Does not] ask for confirmation before initialization.

/REPLACE[:RETAIN]

Detects bad blocks and creates [or retains the old] bad-block replacement table.

/RESTORE

Restores the previous directory on an initialized volume. You can use /RESTORE only if no files have been transferred to the volume since it was initialized.

/SEGMENTS:value

Specifies the number of directory segments to allocate.

Default Directory Sizes

Device	Decimal Number of Segments in Directory
DW (RD50)	16
DW (RD51)	31
DX (RX01)	1
DY (RX02 single-density)	1
DY (RX02 double-density)	4
DZ (RX50)	4
DL (RL01)	16
DL (RL02)	31
DM (RK06/07)	31
DU (Winchester disks)	31
DU (Diskettes)	1
RK (RK05)	16

/VOLUMEID[:ONLY]

Asks for owner name and volume ID during the initialization; with ONLY, writes a new owner name and volume ID without changing the directory.

```
.INITIALIZE/VOLUMEID DU1:
DU1:/Initialize; Are you sure? Y
 Volume ID?  FORTRAN VOL
 Owner?  AJAX-CORP
```

/WAIT

Waits for a volume to be mounted before executing the command.

INSTALL

(KMON)

Enters the device you specify into the monitor's internal tables.

INSTALL device[,device,...]

An example situation in which you might use INSTALL:

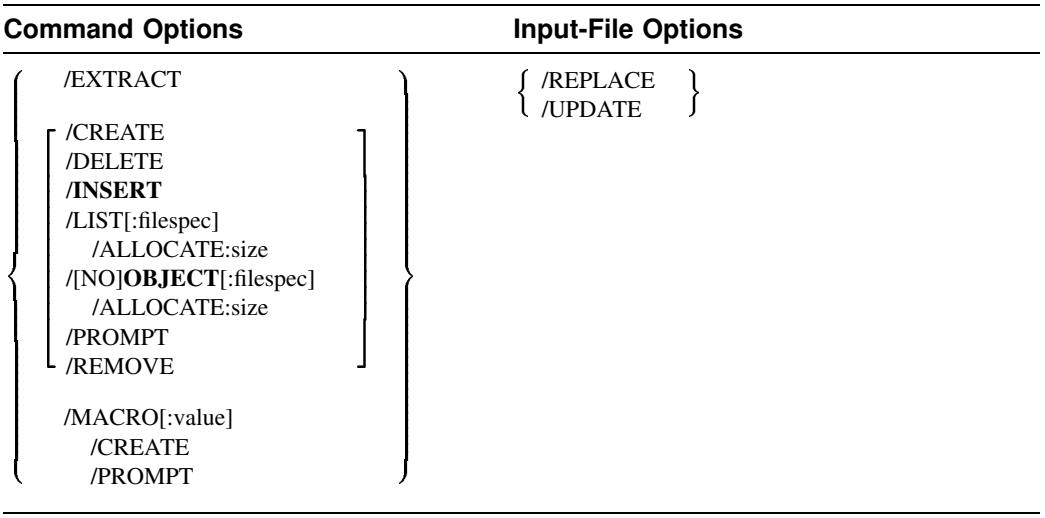
```
.COPY SDH.SYS SD.SYS
.UNLOAD SD
.REMOVE SD
.INSTALL SD
```

LIBRARY

(LIBR)

Lets you create, update, modify, list, and maintain library files.

LIBRARY input-files



/ALLOCATE:size

Reserves space for the output file; a size of -1 reserves the largest possible space. This example uses /ALLOCATE to create the .OBJ library MYLIB.OBJ from the .OBJ library MYFILE.OBJ:

```
LIBRARY/OBJECT:MYLIB/ALLOCATE:-1 MYFILE
```

/CREATE

Creates a new .OBJ library. The following command creates a library called NEWLIB.OBJ from the modules contained in files FIRST.OBJ and SECOND.OBJ:

```
.LIBRARY/CREATE NEWLIB FIRST,SECOND
```

/DELETE

Deletes an .OBJ module from a library. The following example deletes modules SGN and TAN from the library called NEWLIB.OBJ:

```
.LIBRARY/DELETE NEWLIB
Module name? SGN
Module name? TAN
Module name?
```

/EXTRACT

Retrieves an .OBJ module from a library and stores it in a separate file. The following example retrieves module ATAN from NEWLIB.OBJ library and stores it in the file ATAN.OBJ:

```
.LIBRARY/EXTRACT RET
Library? NEWLIB RET
File   ? DU1:ATAN RET
Global ? ATAN RET
Global  ? RET
```

/INSERT

Adds an .OBJ module into an existing library. The following example inserts the modules contained in the files THIRD.OBJ and FOURTH.OBJ into the library called OLDLIB.OBJ:

```
.LIBRARY/INSERT OLDLIB THIRD,FOURTH
```

/LIST[:filespec]

Sends a directory listing of an .OBJ library to the printer or the filespec.

/MACRO[:value]

Creates a macro library. The *value* specifies the size in block of the macro name directory. The following example creates a macro library called NEWLIB.MLB from the ASCII input file SYSMAC.MAC:

```
.LIBRARY/MACRO/CREATE NEWLIB SYSMAC
```

/[NO]OBJECT[:filespec]

[Does not] create a new .OBJ library from an old library; the output file name defaults to the input file name. The following example creates a library called NEWLIB.OBJ, which consists of the library OLDLIB.OBJ plus the modules that are contained in files THIRD.OBJ and FOURTH.OBJ:

```
.LIBRARY/INSERT/OBJECT:NEWLIB OLDLIB THIRD,FOURTH
```

LIBRARY

/PROMPT

Allows more than one line of input files; terminate with two backslashes (//). The following example creates a macro library called MACLIB.MLB from seven input files:

```
.LIBRARY/MACRO/PROMPT MACLIB A,B,C,D RET
*E,F,G RET
*// RET
```

/REMOVE

Deletes global symbols from a library file's directory. The following example deletes the globals GA, GB, GC, and GD from the library OLDLIB.OBJ:

```
.LIBRARY/REMOVE OLDLIB RET
Global? GA RET
Global? GB RET
Global? GC RET
Global? GD RET
Global? RET
```

/REPLACE

Replaces existing modules with modules of the same name. The following example replaces a module SQRT in the library MATHLB.OBJ with a new module, also called SQRT, from the file MFUNCT.OBJ:

```
.LIBRARY MATHLB MFUNCT/REPLACE
```

/UPDATE

Combines the functions of /INSERT and /REPLACE. The following example updates the library OLDLIB.OBJ:

```
.LIBRARY OLDLIB FIRST/UPDATE,SECOND/UPDATE
```

LINK

(LINK)

Combines .OBJ modules into a program that can be run.

LINK filespecs

Command Options

/ALPHABETIZE
/[NO]**BITMAP**
/DEBUG[:filespec]
/DUPLICATE
/[NO]**EXECUTE**[:filespec]
  /ALLOCATE:size
/EXTEND[:value[:type]]
/FILL:value[:type]
/GLOBAL
/IDSPACE
/INCLUDE
/LIBRARY:filespec
/LINKLIBRARY:filespec
/MAP[:filespec] [/ALLOCATE:size | /WIDE]
/PROMPT
/ROUND:value[:type]
/SLOWLY
/STACK[:value[:type]]
/SYMBOLTABLE[:filespec]
/TRANSFER[:value]
{ /BOTTOM:value[:type] | /BOUNDARY[:value[:type]] | /FOREGROUND[:stacksize] | /LDA | /RUN | /TOP:value[:type] | /XM | /LIMIT:value }

/ALLOCATE:size
Reserves space for the output file; a size of -1 reserves the largest possible space.

/ALPHABETIZE
Alphabetically lists the global symbols in the load map.

/[NO]BITMAP
[Does not] create a memory usage bitmap.

/BOTTOM:value[:type]
Specifies the lowest address to be used by the relocatable code in the load module. Invalid with /TOP and /FOREGROUND. The optional type (DAS or INS) is valid only with the /IDSPACE option.

/BOUNDARY[:value[:type]]
Links a specified program section starting at a specified address boundary; invalid with /TOP. The optional type (DAS or INS) is valid only with the /IDSPACE option.

/DEBUG[:filespec]
Links a debugger (ODT, VDT, or user supplied) with your program.

/DUPLICATE
Allows multiple copies of a library or module.

/EXECUTE[:filespec]
Specifies a name for the LINK output file; the output file name defaults to the input file name.

/NOEXECUTE[:filespec]
Suppresses the creation of an output file.

/EXTEND:value[:type]
Extends a program section to a specific octal value. The optional type (DAS or INS) is valid only with the /IDSPACE option.

/FILL:value[:type]
Initializes unused locations in the load module and places a specific octal value in those locations. The optional type (DAS or INS) is valid only with the /IDSPACE option.

/FOREGROUND[:stacksize]
Produces an executable file in relocatable (REL) format for use as a foreground job under a multi-job monitor. The *stacksize* specifies the number of bytes of stack space to allocate for the foreground job.

/GLOBAL

Generates a cross reference of all the global symbols in the load map.

```
.LINK/GLOBAL/MAP:DU1:  MYPROG
```

When you specify a cross-reference listing with the /GLOBAL option, LINK generates the temporary file DK:CREF.TMP. The following command syntax designates a device for this temporary file:

```
.ASSIGN dev: CF
```

/IDSPACE

Causes LINK to generate an extended SAV image file which separates I and D space. /IDSPACE and /FOREGROUND[:stacksize] are incompatible.

/INCLUDE

Lets you take global symbols from any library and include the object modules the symbols specify in the linked memory image.

/LDA

Links files in LDA format; invalid with /FOREGROUND and /XM.

/LIBRARY:filespec

Same as /LINKLIBRARY.

/LIMIT:value

(Valid only with the RSTS/E operating system)

/LINKLIBRARY:filespec

Includes the library file you specify as an object-module library in the linking operation.

/MAP[:filespec]

Sends a load-map listing to the printer or to a filespec. The following command creates a map listing file called MYPROG.MAP on DU1:

```
.LINK/MAP:DU1:  MYPROG
```

/PROMPT

Allows multiple input lines. Terminate with two slashes (//).

```
.LINK/PROMPT COSINE  RET
*TAN/O:1 RET
*COS1/O:1 RET
*SIN3/O:2 RET
*LML3/O:2// RET
```

/ROUND:value[:type]

Rounds the section size to a multiple of the value you specify; the value must be a power of 2. The optional type (DAS or INS) is valid only with the /IDSPACE option.

/RUN

Initiates execution of the resultant SAV file. Invalid with /FOREGROUND and /LDA.

/SLOWLY

Allows the largest possible LINK symbol table.

/STACK[:value]

Sets the stack address in location 42 to the value you specify. When specified with /IDSPACE, if *value* is not specified, the specified stack symbol is verified to be in D-space.

/SYMBOLTABLE[:filespec]

Creates a file containing definitions for all the global symbols in the load module. The following command creates the symbol-table file BTAN.STB:

```
.LINK AOBJ,BOBJ/SYMBOLTABLE:BTAN
```

/TOP:value[:type]

Specifies the highest address to be used by the relocatable code in the load module; invalid with /BOTTOM and /FOREGROUND. The optional type (DAS or INS) is valid only with the /IDSPACE option.

/TRANSFER[:value]

Specifies the starting address for program execution. When specified with /IDSPACE, if *value* is not specified, the specified transfer symbol is verified to be in I-space.

/WIDE

Produces a wide load-map listing.

/XM

Enables special .SETTOP and .LIMIT features provided in a mapped monitor; invalid with /LDA.

LOAD

(KMON)

Copies a device handler from the system device into memory and assigns them to foreground, background, system jobs, or BATCH.

LOAD device[=jobname][,...device[=jobname]]

To assign a device handler for exclusive use by a program, specify the program name. For example, to assign the XL communication-port handler to VTCOM and to assign the LS serial-interface printer handler to SPOOL, do the following:

```
LOAD XL=VTCOM
LOAD LS=SPOOL
```

MACRO

(MACRO)

Runs the MACRO assembler.

MACRO filespec[s])

Command Options	Input-File Options
/CROSSREFERENCE[:type[:type...]]	/LIBRARY
/DISABLE:type[:type...]	
/ENABLE:type[:type...]	
/LIST[:filespec]	
/ALLOCATE:size	
/[NO]**/OBJECT**[:filespec]	
/ALLOCATE:size	
/[NO]SHOW:type[:type...]	

/ALLOCATE:size

Reserves space for the output file; a size of -1 reserves the largest possible space.

/CROSSREFERENCE[:type[:type...]]

Produces a CREF listing; the types are as follows.

Type	Meaning
C	Control sections (.CSECT symbolic names)
E	Error codes
M	Macro symbolic names
P	Permanent symbols (instructions, directives)
R	Register symbols
S	User-defined symbols
None	Equivalent to :S:M:E

/DISABLE:type[:type...]

Specifies a MACRO .DSABL directive; the types are as follows.

Type	Default	What Is Enabled or Disabled
ABS	Disable	Absolute binary output
AMA	Disable	Assembly of all absolute addresses as relative addresses
CDR	Disable	Treating source columns 73 and greater as comments
DBG	Disable	Generation of internal symbol directory (ISD) records during assembly (See Chapter 2 of the *RT–11 Volume and File Formats Manual* for more information on ISD records.)
FPT	Disable	Floating-point truncation
GBL	Enable	Treating undefined symbols as globals
LC	Enable	Accepting lowercase ASCII input
LCM	Disable	Uppercase and lowercase sensitivity of MACRO–11 conditional assembly directives .IF IDN and .IF DIF
LSB	Disable	Local symbol block
PNC	Enable	Binary output
REG	Enable	Mnemonic definitions of registers

/ENABLE:type[:type...]

Specifies a MACRO .ENABL directive; types are listed under /DISABLE.

/LIBRARY

Identifies the file it qualifies as a library file. The following command includes two user libraries (MYLIB1.MLB and MYLIB2.MLB):

```
.MACRO MYLIB1/LIBRARY+A+MYLIB2/LIBRARY+B
```

/LIST[:filespec]

Produces a MACRO assembly listing on the printer or in the specified filespec.

```
.MACRO/LIST MYPROG
```

/[NO]OBJECT[:filespec]

[Does not] generate an .OBJ file; the output file name defaults to the input file name.

/[NO]SHOW:type[:type...]

Specifies MACRO [.NLIST] and .LIST directives; the types are as follows.

Type	Default	Controls
BEX	List	Extended binary code
BIN	List	Generated binary code
CND	List	Unsatisfied conditionals, .IF and .ENDC statements
COM	List	Comments
LD	Nolist	Listing directives with no arguments
LOC	List	Location counter
MC	List	Macro calls, repeat range expansions
MD	List	Macro definitions, repeat range expansions
ME	Nolist	Macro expansions
MEB	Nolist	Macro expansion binary code
SEQ	List	Source line sequence numbers
SRC	List	Source code
SYM	List	Symbol table
TOC	List	Table of contents
TTM	Line printer	Wide or narrow listing format

MOUNT

(LD)

Associates the logical-disk unit (LDn) you specify with the file you specify and optionally assigns it a logical name. This enables you to use a logical disk as if it were a physical disk.

MOUNT logical-disk-unit filespec [logical-device-name]

Command Options

/READONLY
/[NO]WRITE

The MOUNT command associates a logical-disk file with one of eight logical-disk units (LD0–LD7). If you perform a system generation (SYSGEN), and request extended device-unit support, you can access up to 32 logical-disk units at one time.

The following command example associates logical-disk unit 5 (LD5) with the file DATA.DSK on device DU1:

```
.MOUNT LD5: DU1:DATA
```

/READONLY
Does not allow WRITE access through LD to any files contained in the specified logical-disk unit.

/[NO]WRITE
[Does not] make the logical disk write enabled.

PRINT

(PIP, QUEMAN, QUEUE, and SPOOL)

Sends the contents of one or more files to the printer.

PRINT filespec[s]

Command Options

{ [/BEFORE[:date]] }
{ [/SINCE[:date]] }
{ /DATE[:date] }
{ /NEWFILES }
/COPIES:value
/DELETE
/[NO]FLAGPAGE:value
/INFORMATION
/[NO]LOG
/NAME:[dev:]jobname
/OUTPUT:dev:[filespec]
/PRINTER
/PROMPT
/QUERY
/WAIT

/BEFORE[:date]

Prints only those files created before the specified date. The following command prints all MAC files on DU0: created before April 21, 1991:

```
.PRINT/BEFORE:21:APR:91 DU0:*.MAC
```

/COPIES:value

Prints the specified number of files; the valid range is from 1 to 32.

```
.PRINT/COPIES:3 REPORT
```

/DATE[:date]

Prints only those files with the specified creation date.

```
.PRINT/DATE:21:APR:91 DK:*.MAC
```

PRINT

/DELETE

Deletes a file after it is printed.

```
.PRINT/DELETE DU1:PROG1.BAS
```

/[NO]FLAGPAGE:value

(Valid only with SPOOL or QUEUE) [Does not] include the specified number of banner pages at the start of a file. The following example prints three banner pages for each file given:

```
.PRINT/FLAGPAGE:3 PROG1.MAC,PROG1.LST,PROG1.STB
```

/INFORMATION

Displays informational rather than fatal messages for file not found and prints all others.

```
.PRINT/INFORMATION DL0:(FILE1,FILE2,FILE3).TXT
?PIP-I-File not found DL0:FILE2.TXT
```

/[NO]LOG

[Does not] log file names on the terminal as they are printed.

```
.PRINT/LOG/DELETE REPORT
Files copied/deleted:
DK:REPORT.LST to LP:
```

/NAME:[dev:]jobname

Specifies a job name for the files you want printed; with dev, send files to that device.

```
.PRINT/NAME:DU1:JOB5 FILE1, FILE2, FILE3
```

/NEWFILES

Prints only those files that have the current system date.

```
.PRINT/NEWFILES *
Files copied:
DK:OUTFIL.LST   to LP:
DK:REPORT.LST   to LP:
```

/OUTPUT:dev:[filespec]

Specifies the output device for the PRINT command.

/PRINTER

Forces files to be copied directly to the printer if QUEUE is running.

/PROMPT

(Valid only with QUEUE) Allows multiple input lines; terminated with a double slash (//).

```
.PRINT/PROMPT FILE1
*FILE2, FILE3
*FILE4
*FILE5//
```

/QUERY

Asks if specific files should be printed.

```
.PRINT/QUERY *.LST
Files copied:
DK:OUTFIL.LST  to LP:? N
DK:REPORT.LST  to LP:? Y
```

/SINCE[:date]

Prints only those files created on or after the specified date. The following command prints all MAC files on DU0 created on or after April 21, 1991:

```
.PRINT/SINCE:21:APR:91 DU0:*.MAC
```

/WAIT

(Invalid with QUEUE) Waits for the volume to be mounted before executing the command.

PROTECT

(PIP)

Protects a file so that you cannot delete it until you remove the protection.

PROTECT filespec[s]

Command Options

{ [/BEFORE[:date]] }
{ [/SINCE[:date]] }
{ /DATE[:date] }
{ /NEWFILES }
/EXCLUDE
/INFORMATION
/[NO]LOG
/QUERY
/SETDATE[:date]
/SYSTEM
/WAIT

/BEFORE[:date]

Protects only those files created before the specified date. The following command protects all MAC files on DK created before March 20, 1991:

```
.PROTECT/BEFORE:20:MAR:91 *.MAC
Files protected:
DK:A.MAC
DK:B.MAC
DK:C.MAC
```

/DATE[:date]

Protects only those files with the specified creation date. The following command protects all MAC files on DK that were created on March 20, 1991:

```
.PROTECT/DATE:20:MAR:91 *.MAC
```

/EXCLUDE

Protects all the files on a device except the ones you specify. The following command protects all files on DU1 except SAV files and SYS files:

```
.PROTECT/EXCLUDE DU1:*.SAV
?PIP-W-No .SYS action
Files protected:
DU1:ABC.OLD
DU1:AAF.OLD
DU1:COMB.
DU1:MERGE.OLD
```

/INFORMATION

Displays informational rather than fatal messages for file not found and protects all others.

```
.PROTECT/INFORMATION DL0:(FILE1,FILE2,FILE3).TXT
?PIP-I-File not found DL0:FILE2.TXT
```

/[NO]LOG

[Does not] list on the terminal all the files protected by the current command.

/NEWFILES

Protects only the files having the current system date.

```
.PROTECT/NEWFILES DU1:*.BAK
Files protected:
DU1:MERGE.BAK
```

/QUERY

Ask for confirmation before protecting each file. The following example shows querying. Only the file DU1:AAF.MAC is protected:

```
.PROTECT/QUERY DU1:*.*
Files protected:
DU1:ABC.MAC   ? N
DU1:AAF.MAC   ? Y
DU1:MERGE.FOR ? N
```

/SETDATE[:date]

Assigns the specified date to all the newly protected files. The following example protects three files and changes their dates to the current system date:

```
.PROTECT/SETDATE DU1:*.FOR
Files protected:
DU1:ABC.FOR
DU1:AAF.FOR
DU1:MERGE.FOR
```

/SINCE[:date]

Protects only those files created on or after the specified date. The following command protects all MAC files on DU1 that were created on or after April 21, 1991:

```
.PROTECT/SINCE:21:APR:91 DU1:*.MAC
```

/SYSTEM

Protects any unspecified .SYS files in a wildcard operation. This example protects all files on DU1 with the file name MM, including SYS files:

```
.PROTECT/SYSTEM DU1:MM.*
Files protected:
DU1:MM.MAC
DU1:MM.OBJ
DU1:MM.SAV
DU1:MM.SYS
```

/WAIT

Waits for the volume to be mounted before executing the command. The following example protects the file FILE.MAC on an RL02 disk:

```
.PROTECT/WAIT DL0:FILE.MAC
Mount input volume in DL0:; Continue? Y
Mount system volume in DL0:; Continue? Y
```

R

(KMON)

Loads and runs programs from the system device or from the specified file-structured device. See also FRUN, RUN, SRUN, and V/VRUN.

R filespec

Use the R command only with background jobs, including privileged jobs, under a mapped monitor.

REENTER

(KMON)

Starts the program at its reentry address (the start address minus 2).

REENTER

REMOVE

(KMON)

Removes a device name from the monitor's system tables and, under a mapped monitor, can be used to remove a global region.

REMOVE { **xx:[,xx:...]** / **yy[,yy...]** / **region[,region...]** }

where:

xx:	is a device name.
yy	is a device name or global region. (mapped monitors only)
region	is a 1-to-6-character global region. (mapped monitors only)

If you COPY or RENAME a handler, issue a REMOVE and an INSTALL after you change the handler to ensure the system uses the new version of the handler:

```
.COPY SDH.SYS SD.SYS
.UNLOAD SD
.REMOVE SD
.INSTALL SD
```

RENAME

(PIP)

Assigns a new name to an existing file.

RENAME in-filespec[s] out-filespec

Command Options

{ [/BEFORE[:date]] }
{ [/SINCE[:date]] }
{ /DATE[:date] }
{ /NEWFILES }
/INFORMATION
/[NO]LOG
/[NO]PROTECTION
/QUERY
/[NO]**REPLACE**
/SETDATE[:date]
/SYSTEM
/WAIT

/BEFORE[:date]

Renames only those files created before the specified date. The following command renames all MAC files on DU1 created before April 21, 1991:

```
.RENAME/BEFORE:21:APR:91 DU1:*.MAC DU1:*.BAK
```

/DATE[:date]

Renames only those files with the specified creation date. The following command renames all MAC files created on March 20, 1991 to BAK files:

```
.RENAME/DATE:20:MAR:91 DK:*.MAC *.BAK
```

/INFORMATION

Displays an informational rather than a fatal message for files not found and renames all others. In the following example, the input files FILE1.TXT and FILE3.TXT are renamed. However, since RT–11 is unable to find DL0:FILE2.TXT, RT–11 displays a message to inform you:

```
.RENAME/INFORMATION DL0:(FILE1,FILE2,FILE3).TXT
?PIP-I-File not found DL0:FILE2.TXT
```

/[NO]LOG

[Does not] list the renamed files on the terminal.

```
.RENAME DU0:(A*.MAC  *.FOR)
Files renamed:
DU0:ABC.MAC    to DU0:ABC.FOR
DU0:AAF.MAC    to DU0:AAF.FOR
```

/NEWFILES

Renames only those files with the current system date.

/[NO]PROTECTION

[Does not] mark a file as protected.

/QUERY

Asks for confirmation before renaming a file.

```
.RENAME/QUERY  *.BAK  *.MAC
 Files renamed:
DK:PROG1.BAK  to DK:PROG1.MAC ? Y
DK:PROG2.BAK  to DK:PROG2.MAC ? Y
DK:PROG6.BAK  to DK:PROG6.MAC ? Y
DK:LML8A.BAK  to DK:LML8A.MAC ?
DK:LML9 .BAK  to DK:LML9 .MAC ? Y
```

/[NO]REPLACE

[Does not] replace an existing file having the same name.

In the following example, the output file already existed and so nothing is renamed.

```
.RENAME/NOREPLACE DU0:TEST.SAV DU0:DUP.SAV
?PIP-W-Output file found, no operation performed DU0:TEST.SAV
```

/SETDATE[:date]

Assigns the specified date to all the renamed files. The following example renames the specified files and changes their dates to the current system date:

```
.RENAME/SETDATE DU1:(*.FOR *.OLD)
Files renamed:
DU1:ABC.FOR    to DU1:ABC.OLD
DU1:MERGE.FOR  to DU1:MERGE.OLD
```

/SINCE[:date]

Renames all files on or after a specified date. The following command renames only those MAC files on DK created on or after February 24, 1991:

```
.RENAME/SINCE:24:FEB:91 *.MAC *.BAK
Files copied:
DK:A.MAC  to DK:A.BAK
DK:B.MAC  to DK:B.BAK
```

/SYSTEM

Includes SYS files in a wildcard RENAME operation.

This example renames all files on DU1: with the file name MM, including SYS files, to MX files:

```
.RENAME/SYSTEM DU1:MM.* DU1:MX.*
 Files renamed:
DU1:MM.MAC  to DU1:MX.MAC
DU1:MM.OBJ  to DU1:MX.OBJ
DU1:MM.SAV  to DU1:MX.SAV
DU1:MM.SYS  to DU1:MX.SYS
```

/WAIT

Waits for the volume to be mounted before executing the command.

```
.RENAME/WAIT/NOLOG SY:(PRIAM.TXT NESTOR.TXT)
Mount input volume in DU0:; Continue? Y
Mount system volume in DU0:; Continue? Y
```

RESET

(KMON)

Initializes several background system tables and does a general cleanup of the background area.

RESET

Use RESET before you execute a program if a device or the monitor needs reinitialization The following example uses the RESET command before running a program:

```
.RESET
.R MYPROG
```

RESUME

(KMON)

Continues execution of a foreground or system job from the point at which a SUSPEND command was issued.

RESUME [jobname]

The following command resumes execution of the suspended foreground job:

```
.RESUME
```

The next command resumes execution of the suspended system job, QUEUE.SYS:

```
.RESUME QUEUE
```

You can also use RESUME to start a job loaded with FRUN or SRUN using /PAUSE.

RUN

(KMON)

Loads and runs a program from the default device DK, or from the specifed file-structured device. See also FRUN, R, SRUN, and V/VRUN.

RUN filespec [{ **input-list, [output-list]** / **argument** }]

where:

filespec	specifies a SAV executable file.
argument	specifies a single CSI argument to be passed to the program.
input-list	specifies a list of CSI or CCL input file specifications to be passed to the program.
output-list	specifies a list of CSI or CCL output file specifications to be passed to the program.

Examples:

1. RUN executes the program MYPROG.SAV:

```
.RUN DU1:MYPROG
```

2. DU1:*.* is the input and LP:/E is the output:

```
.RUN DIR DU1:*.* LP:/E
```

3. An alphabetical directory listing is displayed on the terminal:

```
.RUN DIR /A
```

SAVE

(KMON)

Writes selected areas of memory to a file.

SAVE filespec address[-address(2)][,address(3)[-address(n)]]

RT–11 transfers memory in 256-word blocks, beginning on boundaries that are multiples of 256_{10}.

The following command saves the contents of locations 10000 through 11777 and 14000 through 14777 and stores them in FILE1.SAV:

```
.SAVE FILE1 10000-11000,14000-14100
```

SET CLI

(KMON)

Enables or disables one or more command-line interpreters.

SET CLI interpreter1,interpreter2,...

Command Summary

SET CLI [[NO]**CCL** | [NO]**DCL** | [**NO**]**UCF** | [NO]**UCL**]

SET CLOCK

(KMON)

Sets the frequency of the system clock.

SET CLOCK frequency

50

Causes the system to use a 50-Hz line frequency clock rate.

60

Causes the system to use a 60-Hz line frequency clock rate (the default).

SET DL

(DL)

Sets the DL disk handler (RL01/RL02 disks).

SET DL condition

CSR=n

Uses *n* as the CSR address for the DL handler.

RETRY=n

Sets the number of retries after an I/O error.

[NO]SUCCES

SUCCES, the default, logs successful I/O transfers as well as errors when running the Error Logger. NOSUCCES logs only I/O errors.

VECTOR=n

Uses *n* as the vector address for the DL handler.

SET DM

(DM)

Sets the DM disk handler (RK06/RK07 disks).

SET DM condition

CSR=n
Uses *n* as the CSR address for the DM handler.

RETRY=n
Sets the number of retries after an I/O error.

[NO]SUCCES
SUCCES, the default, logs successful I/O transfers as well as errors when running the Error Logger. NOSUCCES logs only I/O errors.

VECTOR=n
Uses *n* as the vector address for the DM handler.

SET DU

(DU)

Sets the DU device handler (MSCP disks and diskettes).

SET DU condition

CSR=n
Uses *n* as the CSR address for the first port of the DU MSCP controller.

RETRY=n
Sets the number of retries after an I/O error.

[NO]SUCCES
SUCCES, the default, logs successful I/O transfers as well as errors when running the Error Logger. NOSUCCES logs only I/O errors.

VECTOR=n
Uses *n* as the vector address for the first port of the DU MSCP controller.

SET DUx

(DU)

Defines how an MSCP disk is partitioned.

SET DUx condition

CSRy=n

(Valid only if the handler is assembled for multiple ports) Modifies the DU handler to use *n* as the CSR address for DU controller *y*. The *y* can be from 0 to 3, with 0 being the default. *SET CSR0=n* is equivalent to *SET CSR=n*.

PART=n

Defines the partition of a disk on which device unit *x* resides. The *n* can be from 0 to 255, depending on the size of the disk device (each partition is 64K blocks). The default for *n* is 0.

SET DUx PART=n assigns DUx to disk partition *n*.

PORT=n

Assigns DUx to MSCP port *n*. The *x* can be from 0 to 7, with 0 being the default.

With extended-unit support, *x* can be from 0 to 64_{10}. Specify an extended DU device unit in the format *Dxx*, where *xx* is a number from 0 through 77_8.

The *n* (specifying the port) can be from 0 to 3, with 0 being the default.

UNIT=n

Assigns *DUx* to MSCP unit *n*. The *n* can be from 0 to 251. By default, *n* equals *x*.

VECy=n

(Valid only if the handler is assembled for multiple ports) Modifies the DU handler to use *n* as the vector address for controller *y*. The *y* can be from 0 to 3, with 0 being the default. *SET DU VEC0=n* is equivalent to *SET DU VECTOR=n*.

SET DW

(DW)

Sets the DW disk handler (RDxx disks).

SET DW condition

RETRY=n

Sets the number of retries after an I/O error.

SLOT=n

(Valid only with the PRO Expander) Assigns a copy of the DW handler to the controller for the description for an explanation of how to do this.

[NO]SUCCES

SUCCES, the default, logs successful I/O transfers as well as errors when running the Error Logger. NOSUCCES logs only I/O errors.

[NO]WCHECK

WCHECK verifies output to DW disks by reading data after writing it to the disk. NOWCHECK does not verify output to DW disks.

[NO]WRITE

WRITE write enables DW disk drive unit 0. NOWRITE write locks DW disk drive unit 0.

SET DX

(DX)

Sets the DX diskette handler (RX01 diskettes).

SET DX condition

CSR=n
Uses *n* as the CSR address for the DX handler.

CSR2=n
(Valid only with support for a second controller) Uses *n* as the CSR address for the DX handler.

RETRY=n
Sets the number of retries after an I/O error.

[NO]SUCCES
SUCCES, the default, logs successful I/O transfers as well as errors when running the Error Logger. NOSUCCES logs only I/O errors.

VECTOR=n
Uses *n* as the vector address for the DX handler.

VEC2=n
(Valid only with support for a second controller) Uses *n* as the vector address for the DX handler.

SET DXx

(DX)

Sets the specified unit of the DX diskette handler (RX01 diskettes).

SET DXx condition

[NO]WRITE
WRITE, the default, write enables DX unit *x*. NOWRITE write locks DX unit *x*.

SET DY

(DY)

Sets the DY diskette handler (RX02 diskettes).

SET DY condition

CSR=n
Uses *n* as the CSR address for the DY handler.

CSR2=n
(Valid only with support for a second controller) Uses *n* as the CSR address for the DY handler.

RETRY=n
Sets the number of retries after an I/O error.

[NO]SUCCES
SUCCES, the default, logs successful I/O transfers as well as errors when running the Error Logger. NOSUCCES logs only I/O errors.

VECTOR=n
Uses *n* as the vector address for the DY handler.

VEC2=n
(Valid only with support for a second controller) Uses *n* as the vector address for the DY handler.

SET DYx

(DY)

Sets the specified unit of the DY diskette handler (RX02 diskettes).

SET DYx condition

[NO]WRITE
WRITE, the default, write enables DY unit *x*. NOWRITE write locks DY unit *x*.

SET DZ

(DZ)

Sets the DZ diskette handler (RX50 diskettes for Professional 300 series).

SET DZ condition

RETRY=n
Sets the number of retries after an I/O error.

[NO]SUCCES
SUCCES, the default, logs successful I/O transfers as well as errors when running the Error Logger. NOSUCCES logs only I/O errors.

SET EDIT

(KMON)

Defines an editor as the default editor.

SET EDIT default-editor)

EDIT
is EDIT.SAV.

KED
is KED.SAV (default with distributed unmapped monitors)

KEX
is KEX.SAV (default with distributed mapped monitors)

TECO
is TECO.SAV (TECO is not distributed or supported by Digital.)

SET EL

(KMON)

Sets the Error Logger for single-job monitors.

SET EL condition

[NO]LOG

LOG turns on error logging.

NOLOG, the default, turns off error logging.

PURGE

Clears the internal error-log buffer in a single-job monitor.

SET EL

(KMON)

Sets the Error Logger for single-job monitors.

SET EL condition

[NO]LOG

LOG turns on error logging.

NOLOG, the default, turns off error logging.

PURGE

Clears the internal error-log buffer in a single-job monitor.

SET ERROR

(KMON)

Controls the level of failure that aborts a command file.

SET ERROR condition

ERROR
(the default) Aborts command files if error, fatal, or unconditional errors occur.

FATAL
Aborts command files if fatal or unconditional errors occur.

NONE
Continues processing.

UNCONDITIONAL
Aborts command files if an unconditional error occurs.

WARNING
Aborts command files if warning, error, fatal, or unconditional errors occur.

SET EXIT

(KMON)

Selects whether or not SWAP.SYS is used on job exit.

SET EXIT condition

[NO]SWAP
SWAP, the default, saves the running program in the swap blocks when the program exits so that it can be restarted with the REENTER command.

NOSWAP disables the writing of a program to the swap blocks when the program exits.

SET FORTRA

(KMON)

Sets the default FORTRAN compiler.

SET FORTRA compiler

F4

(the default) Causes the FORTRAN IV compiler to be called by the COMPILE /FORTRAN, EXECUTE/FORTRAN, and FORTRAN commands.

F77

Causes the FORTRAN–77 compiler to be called by the COMPILE/FORTRAN, EXECUTE/FORTRAN, and FORTRAN commands.

SET KMON

(KMON)

Sets KMON to recognize IND (INDirect control-file processor) control files.

SET KMON condition

[NO]IND

IND causes IND.SAV to interpret the expression @*filespec* as a control file to be executed. NOIND, the default, causes KMON to interpret the expression @*filespec* as a command file to be executed.

The syntax *$@filespec* forces NOIND interpretation even when SET IND is in effect.

SET LD

(LD)

Sets the logical-disk subsetting handler.

SET LD condition

CLEAN
Checks all logical-disk assignments to make sure they are valid.

EMPTY
Dismounts all logical-disk assignments.

SET LDx

SET LDx condition

[NO]WRITE
(Used during disk subsetting) WRITE, the default, defines logical-disk unit *x* as being write enabled (read/write access allowed). The *x* can be from 0 to 7. With extended-unit support, *x* can be from 0 to 32_8. The syntax for specifying an extended device unit is **Lxx:**

NOWRITE, used during disk subsetting, defines logical-disk unit *x* as being write locked (read-only access allowed).

SET LP

(LP)

Sets the line-printer handler.

SET LP condition

[NO]BIT8
BIT8 passes the eighth bit unaltered.

NOBIT8, the default, forces the eighth bit to zero.

[NO]CR
CR sends carriage return characters (octal 15) to the line printer.

NOCR, the default, does not send carriage return characters (octal 15) to the line printer.

CSR=n

Usees *n* as the CSR address for the LP handler.

[NO]CTRL

CTRL passes all characters, including nonprinting control characters, to the printer.

NOCTRL, the default, ignores nonprinting control characters.

ENDPAG=n

Appends *n* form feeds at the end of the file. The default is 0.

[NO]FORM

FORM, the default, uses the form-feed character (octal 14) to generate form feeds.

[NO]FORM0

FORM0, the default, issues a form feed before printing block 0.

NOFORM0 does not issue a form feed before printing block 0.

[NO]HANG

HANG, the default, waits for you to make a correction when the line printer is not ready.

NOHANG generates an immediate error when the line printer is not ready.

[NO]LC

Sends lowercase characters to the printer.

NOLC, the default, translates characters in lowercase to uppercase before printing.

LENGTH=n

Defines the length of page by the number of lines. The default length is 66 lines.

SKIP=n

Defines the number of lines to skip at page break; use with SET LP NOFORM. The default is SKIP=0.

[NO]TAB

TAB sends TAB characters to the line printer.

NOTAB, the default, expands TAB characters by sending multiple spaces to the line printer.

VECTOR=n

Uses *n* as the vector of the line printer controller.

WIDTH=n

Sets the line width to *n*, where *n* can be from 30_{10} to 255_{10}. The default is 132.

SET LS

(LS)

Sets the serial-printer handler.

SET LS condition

[NO]BIT8

Passes the eighth bit unaltered.

NOBIT8, the default, forces the eighth bit to zero.

[NO]CR

CR, the default, sends carriage-return characters (octal 15) to the printer.

NOCR does not send carriage-return characters (octal 15) to the printer.

CSR=n

Uses *n* as the CSR address for the LS handler.

SET LS

[NO]CTRL

CTRL, the default, passes all characters, including nonprinting control characters, to the printer.

NOCTRL ignores nonprinting control characters.

ENDPAG=n

Appends *n* number of form feeds at the end of each file. The default is 0.

[NO]FORM

FORM, the default, uses the form-feed character (octal 14) to generate form feeds.

NOFORM use multiple line-feed characters to simulate form feeds.

[NO]FORM0

FORM0, the default, issues a form feed before printing block 0.

NOFORM0 does not issue a form feed before printing block 0.

[NO]GRAPH

GRAPH sets the LS handler to output all characters regardless of the width of the line.

NOGRAPH, the default, sets the LS handler to not print any characters that do not fit on a line.

[NO]HANG

HANG, the default, waits for you to make a correction when the printer is not ready.

NOHANG generates an immediate error when the printer is not ready.

[NO]LC

LC, the default, sends lowercase characters to the printer.

NOLC translates lowercase characters to uppercase before printing.

LENGTH=n

Defines length of page; the initial value of *n* is 66 lines.

LINE=n

(With multiterminal support only) Modifies the serial-printer handler to use line *n* of a multiterminal monitor as the serial port.

[NO]MTTY

(With multiterminal support only) MTTY selects use of multiterminal as the interface to the serial printer. NOMTTY selects use of a DL interface for which it was built as the interface to the serial printer.

SKIP=n

Defines the number of lines to skip at page break; use with SET LIS NOFORM. The default is SKIP=0.

SPEED=n

(Valid for a Professional 300 series computer only) Sets the printer to run at baud *n*, where *n* can be any of the following baud rates:

```
 50  1200
 75  1800
110  2000
134  2400
150  3600
200  4800
300  9600
600  19200
```

[NO]TAB

TAB sends TAB characters (octal 11) to the printer.

NOTAB, the default, simulates tabs with spaces.

VECTOR=n

Uses *n* as the vector address of the LS handler.

WIDTH=n

Sets the line width to *n*, where *n* is a decimal integer between 30 and 255 inclusive. The initial value of *n* is 132.

SET MODE

(KMON)

Sets or clears the obsolete FBMON$ bit in the configuration offset of the monitor fixed area to indicate the obsolete SJ monitor characteristics.

SET MODE [NO]SJ

[NO]SJ

SET MODE SJ turns the FBMON$ bit off.

SET MODE NOSJ, the default, turns the FBMON$ bit on.

SET MM

(MM)

Sets the MM magtape handler (TJU16/TJE16/TU45/TJU77 magtapes).

SET MM condition

CSR=n

Uses *n* as the CSR address for the MM handler.

DEFALT=9

Sets the default settings for 9-track magtape. The 9-track defaults are:

DENSE=809
ODDPAR

DENSE=density

Sets density for the 9-track tape handler. The density can be:

800 (or 809) bpi
1600 bpi

[NO]ODDPAR

ODDPAR, the default, sets parity to odd for 9-track tape.

NOODDPAR sets parity to even for 9-track tape.

RETRY=n

Specifies the number of times the device handler attempts to recover from an I/O error.

VECTOR=n

Uses *n* as the vector address for the MM handler.

SET MS

(MS)

Sets the MS magtape handler (TK25/TS11/TSV05/TSU05/TU80 magtapes).

SET MS condition

CSR=n

Uses *n* as the CSR address for the first controller. *SET CSR=n* is equivalent to *SET CSR1=n*.

CSRx=n

(Valid only with multiple ports) Uses *n* as the CSR address for MS controller *x*; *x* can be from 2 to 8.

RETRY=n

Specifies the number of times the device handler attempts to recover from an I/O error.

VECTOR=n

Uses *n* as the vector address for the first controller. *SET MS VECTOR=n* is equivalent to *SET MS VEC1=n*.

VECx=n

(Valid only with multiple ports) Uses *n* as the vector address for controller *x*; *x* can be from 2 to 8.

SET MT

(MT)

Sets the MT magtape handler (TM11/TMA11/TS03/TE16 magtapes).

SET MT condition

CSR=n

Uses *n* as the CSR address for the MT handler.

DEFALT=n

Sets defaults for 7- or 9-track magtape; *n* can be 7 or 9.

DENSE=n

Sets density for 7- or 9-track magtape; *n* can be 200, 556, **800**, 807, or 809 bpi.

DUMP

Writes bytes to 7-track magtape at 800 bpi.

[NO]ODDPAR

ODDPAR, the default, sets odd parity for 7- or 9-track magtape.

NOODDPAR sets even parity for 7- or 9-track magtape.

RETRY=n

Specifies the number of times the device handler attempts to recover from an I/O error.

VECTOR=n

Uses *n* as the vector address for the MT handler.

SET MU

(MU)

Sets the MU magtape handler (TMSCP magtape devices).

SET MU condition

CSR=n

Uses *n* as the CSR address for the first controller.

RETRY=n

Specifies *n* as the number of times MU attempts to recover from an error.

[NO]SUCCES

(Valid only with error-logging support) SUCCES, the default, modifies the MU handler to log successful I/O transfers as well as errors when the Error Logger is running.

NOSUCCES, the default, modifies the MU handler to not log successful I/O transfers as well as errors when the Error Logger is running.

VECTOR=n

Uses *n* as the vector for the first controller.

SET MUx

(MU)

Sets the specified unit of the MU magtape handler (TMSCP magtape devices).

SET MUx condition

CSRy=n

(Valid only if you sysgen the specified MU controller handler) Modifies the MU device unit *y* to use *n* as the CSR address for a controller. *y* can be 0, 1, 2, or 3. *SET MU CSR0=n* is the same as *SET MU CSR=n*.

DEFALT

Returns magtape unit *n* to the default density of 6250 bpi. Valid only with 9-track TMSCP magtapes and sets only the specified magtape unit.

DENSE=n

Sets density for only 9-track TMSCP magtapes and for only the specified magtape unit. *n* can be 1600 or 6250. Setting DENSE=1600 specifies 1600 bpi (phase encoded). Setting DENSE=6250 specifies 6250 bpi (group code recording).

PORT=n

(Valid only if the handler is assembled for multiple ports) Defines which port to access when magtape unit *x* is specified. The *n* can be from 0 to 3 and corresponds to the variable *y* used with the SET MU CSRy=n and SET MU VECy=n commands. By default, the port variable *n* is the same as the magtape unit *x*.

UNIT=n

Defines which TMSCP unit number to access when magtape unit *x* is specified. The variable *n* can be from 0 to 255; the default is 0.

VECy=n

(Valid only if you sysgen the specified MU controller handler) Modifies the MU device unit *y* to use *n* as the vector for a controller. *y* can be 0, 1, 2, or 3. *SET MU VECTOR=n* is the same as *SET MU VEC0=n*.

SET NC

(NC)

Sets the Ethernet handler for the Professional 300 series processor.

SET NC condition

SHOW

Displays the station physical address for the DECNA controller.

SET NL

(NL)

Sets the NULL handler.

SET NL condition

[NO]SYSGEN

SYSGEN causes NL to match the current monitor's SYSGEN parameters.

NOSYSGEN inhibits NL from installing under the current monitor.

[NO]TOY

SET NL TOY, the default, runs the NL handler installation code. This code transfers date and time information from the KDJ11–E Time-Of-Year (TOY) clock to the resident monitor.

SET NL NOTOY prevents the NL handler from transferring the current date and time to the resident monitor.

SET NQ

(NQ)

Sets the Ethernet handler for Q-bus processors.

SET NQ condition

CSR=n
Uses *n* as a CSR address for the DEQNA or DELQA controller.

SHOW
Displays the station physical address for the DEQNA or DELQA controller.

VECTOR=n
Uses *n* as a vector address for the DEQNA or DELQA controller.

SET NU

(NU)

Sets the Ethernet handler for UNIBUS processors.

SET NU condition

CSR=n
Uses *n* as the CSR address.

SHOW
Displays the station physical address for the DEUNA or DELQA controller and also identifies the controller type.

VECTOR=n
Uses *n* as the vector address.

SET RK

(RK)

Sets the RK device handler (RK05 devices).

SET RK condition

CSR=n
Uses *n* as the CSR address for the RK handler.

RETRY=n
Specifies the number of times the device handler attempts to recover from an I/O error.

[NO]SUCCES
SUCCES, the default, logs successful I/O transfers as well as errors when running the Error Logger. NOSUCCES logs only I/O errors.

VECTOR=n
Uses *n* as the vector address for the RK handler.

SET RUN

(VBGEXE)

Enables or disables VBGEXE (the virtual run utility) to automatically load and execute programs in a completely virtual environment. See also V/VRUN.

SET RUN condition

[NO]VBGEXE
VBGEXE enables VBGEXE to automatically load and execute programs under extended-memory monitors.

NOVBGEXE disables VBGEXE to automatically run programs.

SET SD

(SD)

Configures the DBG–11 SD handler.

SET SD condition

See the *DBG–11 Symbolic Debugger User's Guide* for the listing and description of the SET SD commands.

SET SL

(SL)

Sets the single-line (SL) command editor.

SET SL condition

ASK
Automatically checks for the type of terminal.

[NO]KED
KED turns on keypad editing functionality.

NOKED, the default, turns off keypad editing.

KMON
Enables use of SL by KMON but not by user programs.

[NO]LEARN
LEARN leaves help text on the screen and scrolls input below it.

NOLEARN, the default, does not keep help text on the screen.

OFF
Disables the single-line editor. This is the default.

ON
Enables the single-line editor for use both by KMON and by user programs and loads the SL handler.

[NO]LET

LET lets you define a character or a program function key to substitute for a command string.

NOLET, the default, disables the LET utility.

[NO]RECALL

(only with mapped monitors) RECALL lets you use the RECALL command to recall previously entered commands.

SET NORECALL, the default, disables the use of the RECALL command.

[NO]SYSGEN

SYSGEN configures the SL handler to match the current monitor's SYSGEN parameters.

NOSYSGEN inhibits SL from installing under the current monitor.

[NO]TTYIN

TTYIN enables single-line editing of .TTYIN input.

NOTTYIN disables single-line editing of .TTYIN input.

VTxxx

Configures the SL handler to use the type of terminal you specify (VT100 or VT102).

WIDTH=n

Sets the line width of the display to *n* number of characters. *n* can be from 30 to 132_{10}; the initial value of *n* is 79.

SET SP

(SP)

Sets the spool handler.

SET SP condition

EXIT

EXIT stops SPOOL in a synchronous manner. Use SET SP EXIT to stop SPOOL from within a command file so that the monitor prompt is not returned until all SPOOL activity is terminated.

SET SPx

(SP)

Sets the spool handler for the specified device or for all spooled devices if no device is specified.

SET SPx condition

ENDPAG=n

Appends *n* form feeds at the end of each file sent to spooled device *x*.

ENDPAG=0 does not append form feeds at the end of a file.

FLAG=n

Specifies the number of flag pages to be prefixed to each file sent to spooled device *x*.

[NO]FORM0

FORM0 issues a form feed on spooled device *x* each time SPOOL encounters block 0 of a file to be printed.

NOFORM0, the default, turns off FORM0 mode for spooled device *x*.

KILL

Aborts output to the *x* spooled device.

NEXT

Stops printing the current file on spooled device *x* and proceeds to the next file queued to that device.

[NO]WAIT

WAIT interrupts output to spooled device *x*, while output to any other spooled device continues.

NOWAIT resumes sending spooled output to device *x* after output to that device has been interrupted.

[NO]WIDE

WIDE causes SPOOL to generate 132-column flag pages for specified device *x*.

NOWIDE causes SPOOL to generate 80-column flag pages for specified device *x*.

SET ST

(ST)

Makes the SYSGEN configuration bits of the ST handler compatible with the currently running monitor.

SET ST SYSGEN

SET TT (TERMINAL)

(TT)

Sets the terminal handler. SET TERMINAL can be substituted for SET TT.

SET TT condition

CONSOL=n
(Requires multiterminal support) Changes the background console terminal to the terminal defined as logical terminal *n*.

[NO]CRLF
(Valid for multi-job monitors only) CRLF, the default, inserts a RETURN/LINE FEED when you attempt to type past the right margin.

NOCRLF takes no special action at the right margin.

[NO]FB
(Valid for multi-job monitors only) FB, the default, treats ^B (CTRL/B) and ^F (CTRL/F) (and ^X (CTRL/X) in monitors that include system job support) as program control characters.

NOFB assigns no special meaning to ^B (CTRL/B), ^F (CTRL/F), and ^X (CTRL/X).

[NO]FORM
(Valid for multi-job monitors only) FORM indicates that the console terminal has hardware form feed capability.

NOFORM, the default, simulates form feeds with eight line feeds.

[NO]PAGE
(Valid for multi-job monitors only) PAGE, the default, interprets ^S (CTRL/S) as XOFF and ^Q (CTRL/Q) as XON to stop and start terminal output, respectively.

NOPAGE causes ^S (CTRL/S) and ^Q (CTRL/Q) to have no special meaning.

[NO]QUIET

QUIET disables the echoing of lines from command files.

NOQUIET, the default, enables echoing of command files as they are processed.

[NO]SCOPE

SCOPE, the default, echoes RUBOUT characters as backspace-space-backspace.

NOSCOPE echoes RUBOUT characters by enclosing the deleted characters in backslashes.

[NO]TAB

(Valid for multi-job monitors only) TAB indicates that the console terminal has hardware tab stops.

NOTAB, the default, simulates tab stops every eight positions.

WIDTH=n

(Valid for multi-job monitors only) Sets the terminal width to *n*, where *n* is an integer between 30 and 255_{10}; the initial value of *n* is 80.

SET UB

(UB)

Sets the UB handler that supports UNIBUS mapping registers (UMR) hardware in UNIBUS processors.

SET UB condition

[NO]INSTAL

INSTAL, the default, allows installation of the UB handler with valid UNIBUS configurations.

NOINSTAL prevents installation of the UB handler with valid UNIBUS configurations.

PERM=decimal-value

Sets the number of permanent UMRs to reserve for handlers that are not installed during the boot process.

[NO]SERIAL=octal-value

SERIAL=octal-value forces serial satisfaction of outstanding I/O requests for the specified job, where *octal-value* is the job number.

NOSERIAL=octal-value, the default, satisfies outstanding I/O requests for the specified job, where *octal-value* is the job number, as required UMRs are free. I/O requests are satisfied as quickly as possible and not necessarily in serial order.

SET USR

(KMON)

Sets the state of USR (User Service Routine) swapping.

SET USR condition

[NO]SWAP

SWAP, the default, allows the background job to swap the USR. This setting is not valid for mapped monitors, but is the default mode for unmapped monitors.

NOSWAP locks the USR in memory so that it cannot be swapped.

SET VM

(VM)

Sets the VM (Virtual Memory) handler.

SET VM condition

BASE=nnnnnn

Lets you select the location in memory where block 0 of a virtual disk will begin (the base address). The VM handler uses memory at and above *nnnnnn*, a physical address multiplied by 100_8.

[NO]INSTAL

INSTAL, the default, causes the VM handler to install at the system boot.

NOINSTAL causes an otherwise valid VM handler to not install at the system boot.

SIZE=nnnnnn

Lets you select the memory size in blocks (512 bytes for each block) of the virtual device. If you SET VM SIZE=0, VM allocates all available memory from the SET VM BASE value to the top of physical memory. VM SIZE=0 is the default.

SET WILD

(KMON)

Sets the monitor to recognize wildcards in file specifications.

SET WILD condition

EXPLICIT

The system recognizes file specifications exactly as typed.

IMPLICIT

The system fills in missing fields in file specifications with wildcards (asterisks). This is the default.

SET XC

(XC)

Sets the baud rate of the communication-port handler for Professional 300 series processor.

SET XC condition

SPEED=n

Sets the communication port to run at baud *n*, where *n* can be:

```
50     1200
75     1800
110    2000
134    2400
150    3600
200    4800
300    9600
600    19200
```

SET XL

(XL)

Sets the communication-port handler for PDP–11 processors.

SET XL condition

LINE=n

(Valid with multiterminal support only) Uses *n* as the multiterminal serial port for XL. The value of *n* can be between 0 and 16; the default is 1.

CSR=n

Uses *n* as the CSR address for the XL handler.

[NO]MTTY

(Valid with multiterminal support only) MTTY selects use of multiterminal as the interface to the XL handler. NOMTTY selects use of a DL interface for which it was built as the interface to the XL handler.

VECTOR=n

Uses *n* as the vector address for the XL handler.

SETUP

(SETUP)

Sets hardware characteristics for the terminal, printer, and system clock.

SETUP [hardware] mode[,mode...]

Table 3–2: Summary of SETUP Modes

Mode	Type	Description
132COLUMNS	Terminal	Displays 132 characters across screen.
80COLUMNS	Terminal	Displays 80 characters across screen.
12HOUR	Clock	Puts the system clock into 12-hour mode (Professional only).
24HOUR	Clock	Puts the system clock into 24-hour mode (Professional only).
480INTERLACE	Terminal	Sets the PRO380 to 480 INTERLACE mode (Professional 380 only).
ANSWERBACK:/text/	Terminal	Places text into the ANSWERBACK buffer (Professional only).
[NO]BCD	Clock	Converts the system clock to BCD [binary] mode (Professional only).
BLOCK	Terminal	Displays the cursor as a block, the default (Professional only).
[**NO**]**BOLD**	Terminal/Printer	Enables [disables] bold printing. NOBOLD is the default.
CAPS	Terminal	Causes LOCK to function as a VT100 key (Professional only). This is the default.
CLEAR	Terminal/Printer	Clears the screen and performs a soft reset.
[NO]**CLICK**	Terminal	Causes keys to [not] click when pressed (Professional only). CLICK is the default.
CLOCK SHOW	Clock	Displays the current clock settings (Professional only).

Table 3–2 (Cont.): Summary of SETUP Modes

Mode	Type	Description
COLOR	Terminal	Informs the software that terminal is a color terminal (Professional color terminals only).
COLOR1 ON COLOR2	Terminal	Sets the text to color1 and background to color2; colors are black, blue, cyan, green, magenta, red, white, and yellow (Professional color terminals only).
[**NO**]**COMPOSE**	National Replacement	Enables [disables] compose character key (Professional only).
[NO]**CURSOR**	Terminal	Causes the cursor to be [in]visible (Professional only).
DARK	Terminal	Displays light characters on a dark background.
DATA	National Replacement	Sets NRC language for data processing characters (Professional only).
DATE:dd:mmm:yy	Clock	Sets the system date as specified.
DEFAULTS	Control	Returns the terminal to its factory settings (Professional terminal only).
DENSE	Terminal/Printer	Overlaps two sets of dots for a close approximation of letter-quality printing.
DOWN	Terminal/Printer	Moves the print down 1/2 inch for alignment.
DRAFT	Printer	Sets print to an 8-x-9 dot matrix.
[NO]DST	Clock	Puts the system clock into [non]daylight savings time mode (Professional only).
GENERIC100	Terminal	Sets the reply to a device ID query as a generic VT100.
GRAPHIC	Terminal/Printer	Enters GRAPHIC mode.
HELP	Control	Lists brief descriptions of the SETUP commands.
HORIZONTAL:n	Terminal/Printer	Prints *n* characters for each inch.
[**NO**]**INTERLACE**	Terminal	Enters [exits] INTERLACE mode.

Table 3–2 (Cont.): Summary of SETUP Modes

Mode	Type	Description
JUMP	Terminal	Scrolls lines across screen as fast as received.
LANDSCAPE	Printer	Sets printer for wider page dimension (LN03 laser printers only).
LANG:xxx	Terminal/Printer	Enables the font and for the specified country code (Professional only).
LETTER	Terminal/Printer	Enables enhanced density and 10 characters per inch.
LIGHT	Terminal	Enables the display of dark characters on a light background.
LISTING	Terminal/Printer	Enables normal density and 16.5 characters per inch.
[**NO**]**MARGINBELL**	Terminal	Enables [disables] bell 8 columns from right margin (Professional only).
MEMO	Terminal/Printer	Enables memo quality printing.
MONO	Terminal	Informs software that the terminal is a black and white terminal (Professional only).
[**NO**]**NEWLINE**	Terminal	Enables [disables] generation of ⌐ LF when RET is typed.
NORMAL	Terminal/Printer	Turns off terminal and printer character attributes, such as REVERSE, UNDERLINE, and BOLD.
PAGELENGTH:n	Terminal/Printer	Prints *n* lines per page.
PORTRAIT	Printer	Sets the printer for narrow page dimension (LN03 laser printers only).
[NO]**REPEAT**	Terminal	Enables [non]repetition of characters when the key is held down (Professional terminal only).
RESET	Control	Resets the terminal to previously saved characteristics.
[NON]REVERSE	Terminal	Places terminal characters in reverse [normal] video mode.

Table 3–2 (Cont.): Summary of SETUP Modes

Mode	Type	Description
RETAIN	Control	Splits off unused NRC languages and retains selected language (Professional terminal only).
SAVE	Control	Saves terminal characteristics.
SETCOLOR color [red:value,blue:value, green:value]	Terminal	Customizes text and background colors (Professional 350 and 380 only).
SETCOLOR color FACTORY	Terminal	Sets the specified color to the distributed setting (Professional only).
SHIFT	Terminal	Causes LOCK to function as a typewriter key (Professional only).
SHOW	Control	Shows current settings. (Professional only)
SMOOTH	Terminal	Scrolls smoothly and slowly up the screen.
[NO]TABS[:n:n:n]	Terminal/Printer	Sets [clears] tabs at specified columns.
TEXT	Terminal/Printer	Exits GRAPHIC mode.
TIME:hh:mm:ss	Clock	Sets the system time as specified.
TYPE	National Replacement	Sets NRC language for typewriter characters (Professional only).
[**NO**]**UNDERLINE**	Terminal/Printer	Enables [disables] use of underlined printing.
UNDERSCORE	Terminal	Displays the cursor as an underscore (Professional only).
UP	Terminal/Printer	Moves the print up 1/2 inch for alignment.
VERTICAL:n	Terminal/Printer	Prints *n* lines per inch.
VT100	Terminal	Sets terminal into VT100 specific mode.
[**NO**]**WRAP**	Terminal/Printer	Enables [disables] wrapping around to the following line.

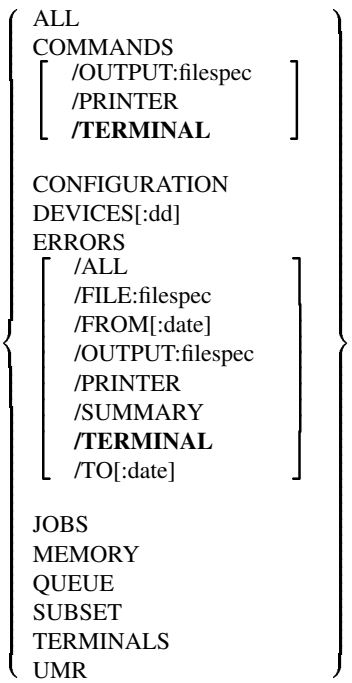
SHOW

(RESORC)

Lists on the console terminal information about the monitor configuration and the available hardware.

SHOW option

Command Options



ALL

Is a combination of the CONFIGURATION, DEVICES, device assignments (SHOW with no option), JOBS, TERMINALS, MEMORY, and SUBSET options in that order.

SHOW

COMMANDS

Lists the contents of the UCL data file on the terminal.

```
.SHOW COMMANDS
!                           User Command Linkage (UCL)

      ALL               :== SHOW ALL\TIME\DATE\\
      FILES             :== DIRECTORY/VOLUME/COLUMNS:1 *.TXT
      V                 :== VBGEXE
      VM                :== $@VM\BOOT VM:\\
```

COMMANDS/OUTPUT:filespec

Sends contents of the UCL data file to the specified file.

COMMANDS/PRINTER

Sends contents of the UCL data file to the printer.

CONFIGURATION

Lists the monitor version, parch level, monitor SET options in effect, hardware configuration, and special features.

DEVICES[:dd]

Lists the available device handlers, status, and vector addresses; with dev lists information about only that device.

ERRORS

(Valid only if you have error logging enabled) Provides information about I/O transfers.

ERRORS/ALL	lists all errors that have occurred.
ERRORS/FILE:filespec	lists the error log from a file.
ERRORS/FROM:date	lists errors that occurred on or after the specified date.
ERRORS/TO:date	lists all errors that occurred before the date (specified in the format *dd:mmm:yy*).
ERRORS/OUTPUT:filespec	writes the error report to a file.
ERRORS/PRINTER	prints the error report on the printer.
ERRORS/SUMMARY	lists I/O errors only.
ERRORS/TERMINAL	displays the error log on the console terminal.

JOBS

Lists job names and numbers, assigned console, priority, state, and memory limits.

```
.SHOW JOBS

JOB  NAME    CONSOLE  LEVEL  STATE    LOW     HIGH    IMPURE
---  ----    -------  -----  -----    ---     ----    ------
14   QUEUE     0       6     SUSPEND  116224  130306  115254
 0   RESORC    0       0     RUN      000000  126110  132344
```

MEMORY

Lists the organization of physical memory, locations of jobs and device handlers, and the base addresses of the USR and KMON.

The following example shows the organization of physical memory when running under the XM monitor:

```
.SHOW MEMORY

------- Extended Memory --------
Address   Module    Words   Type
-------   ------    -----   ----
17760000  IOPAGE     4096.   PRM    HDW    BYP
02000000  MEMTOP
01000000  VM       131072.   SHR
00220700  ......    93984.
00206300  SL         2688.   PVT
00200200  MU         1568.   PVT
00171700  SP         1632.   SHR
00165000  DU         1248.   PVT
00160000  MCA        1280.   PRM
00000000  KERNEL    28672.   PRM    HDW

------ Low Memory -------
Address   Module    Words
-------   ------    -----
157340    DU          144.
117206    RMON       8237.
105144    USR        2577.
001000    ..BG..    17458.
```

QUEUE

(Valid only when QUEUE or SPOOL is running) Lists the contents of the print queue.

SHOW

SUBSET

Displays the subsetting of physical disks into logical disks. For example, if you mount a 1000-block file DU4:WRK.DSK on logical device unit LD0 and assign it the name WRK, SHOW SUBSET displays:

```
LD0 is DU4:WRK.DSK[1000.] = WRK
```

TERMINALS

Lists the status of any active terminals; valid only with multiterminal support.

```
.SHOW TERMINALS
Unit Owner     Type       WIDTH TAB CRLF FORM SCOPE SPEED
-------------------------------------------------------
  0          Console    DL   80  No  No   No    Yes   N/A
  1          Local      DL   80  No  No   No    Yes   N/A
 10          Local      DH   80  No  No   No    Yes  9600
 11          Local      DH   80  No  No   No    Yes  9600
 12          Remote     DH   80  No  No   No    Yes  9600
```

UMR

Displays information about the UMRs (UNIBUS mapping registers) if the UB pseudohandler is loaded, or information about why UB is not loaded.

```
SHOW UMR
                UMR allocation
-------------------------------------------------
00 ...... 1  10 ...... 2  20 ...... 9  30 ..MS.. P
01 ...... 1  11 ...... 2  21 ...... 9  31 ...... P
02 ...... 1  12 ...... 2  22 ...... 9  32 ...... P
03 ...... 1  13 ...... 4  23 ...... 9  33 ...... P
04 ...... 1  14 ...... 4  24 ...... 9  34 ...... P
05 ...... 2  15 ...... 4  25 ...... 9  35 ..DU.. P
06 ...... 2  16 ...... 4  26 ...... 9  36 ..DU.. P
07 ...... 2  17 ...... 9  27 ...... 9  37 IOPAGE P

2. UMRs in use
2. UMRs permanently assigned
0. UMRs dynamically assigned

0. requests waiting for UMR allocation

RESORC = NOSERIAL
```

SQUEEZE

(DUP) Consolidates directory entries. Consolidates all unused blocks and directory entries on a volume.

SQUEEZE device

Command Options

/OUTPUT:device
/[NO]**QUERY**
/WAIT

/OUTPUT:device

Copies the files from the input volume to the output volume specified in *device*. The following command copies all the files from DU0: to DU1: in compressed format, leaving DU0 unchanged:

```
.SQUEEZE/OUTPUT:DU1: DU0:
```

/[NO]QUERY

/QUERY [Does not] ask for confirmation before performing the SQUEEZE operation.

/WAIT

Waits for the volume to be mounted before executing the command.

SRUN

(KMON)

Initiates system jobs. See also FRUN, R, RUN, and V/VRUN.

SRUN filespec

Filespec Options

/BUFFER:value
/LEVEL:value
/NAME:[job-number/job-name]
/PAUSE
/TERMINAL:value

/BUFFER:value

Allocates the specified number of words of memory in addition to the actual program size.

/LEVEL:value

Assigns an execution priority level to the job, where *value* can be 1 through 6.

/NAME:job-number/job-name

Assigns a *job-number* or a logical *job-name* to a program.

The following commands run VTCOM as a system job and assign it the number 6. All further references to VTCOM must use 6, and that number/name assignment remains in effect until you reboot your system.

```
SRUN SY:VTCOM.SAV/NAME:6/PAUSE
LOAD XC=6
RESUME 6
```

/PAUSE

Prints the load address of a program and waits for the RESUME command.

/TERMINAL:value

(Requires multiterminal support) Assigns the specified terminal to the system job.

START

(KMON)

Begins execution of a program at the specified address.

START [address]

SUSPEND

(KMON)

Temporarily stops execution of the foreground or system job.

SUSPEND [jobname]

The following command suspends execution of the system job, QUEUE, that is currently running on a system having system job support:

```
.SUSPEND QUEUE
```

TIME

(KMON)

Sets or displays the time of day.

TIME [hh:mm:ss]

where:

hh	specifies the hour (from 0 to 23).
mm	specifies the minutes (from 0 to 59).
ss	specifies the seconds (from 0 to 59).

The following command enters the time, eleven fifteen in the morning:

```
.TIME 11:15
```

The next command displays the current time:

```
.TIME
11:15:01
```

TYPE

(PIP)

Displays the contents of one or more files on the terminal.

TYPE filespec[s]

Command Options

{ [/BEFORE[:date]] }
{ [/SINCE[:date]] }
{ /DATE[:date] }
{ /NEWFILES }
/COPIES:value
/DELETE
/INFORMATION
/[NO]LOG
/QUERY
/WAIT

/BEFORE[:date]

Displays on the terminal only the files created before the specified date.

```
.TYPE/BEFORE:24:MAR:91 *.MAC
```

/COPIES:value

Displays the specified copies of the file; the number can be from 2 to 32.

```
.TYPE/COPIES:3 REPORT
```

/DATE[:date]

Displays on the terminal only those files with the specified creation date.

```
.TYPE/DATE:20:MAR:91 DK:*.MAC
```

/DELETE

Deletes a file after it is displayed.

/INFORMATION

Displays informational rather than fatal messages for files not found and displays the contents of all others.

/[NO]LOG

[Does not] list names of files as they are displayed.

/NOLOG prevents a list of the displayed files from displaying on the terminal.

/NEWFILES
Displays only those files that have the current date.

/QUERY
Asks for confirmation before displaying a file.

/SINCE[:date]
Displays on the terminal only the file created on or after the specified date. The following command displays only those MAC files on DK created on or after April 21, 1991:

```
.TYPE/SINCE:21:APR:91 *.MAC
```

/WAIT
Waits for the volume to be mounted before executing the command.

UNLOAD

(KMON) Removes previously loaded handlers and terminated foreground or system jobs from memory.

UNLOAD { **device[,device,...]** / **jobname[,jobname,...]** }

The following command releases the line printer and RL02 handlers and frees the area they previously held:

```
.UNLOAD LP:,DL:
```

UNPROTECT

(PIP)

Removes a file's protected status so that it can be deleted.

UNPROTECT filespec[s]

UNPROTECT

Command Options

{ [/BEFORE[:date]] }
{ [/SINCE[:date]] }
{ /DATE[:date] }
{ /NEWFILES }
/EXCLUDE
/INFORMATION
/[NO]LOG
/QUERY
/SETDATE[:date]
/SYSTEM
/WAIT

/BEFORE[:date]

Removes protection from only those files created before the specified date. The following command removes the protected status of all MAC files on DK created before March 20, 1991:

```
.UNPROTECT/BEFORE:20:MAR:91 *.MAC
Files unprotected:
DK:A.MAC
DK:B.MAC
DK:C.MAC
```

/DATE[:date]

Removes protection from only those files with the specified creation date. The following command removes the protected status of all MAC files on DK created on March 20, 1991:

```
.UNPROTECT/DATE:20:MAR:91 *.MAC
```

/EXCLUDE

Removes protection from all the files except those you specify.

/INFORMATION

Displays informational rather than fatal messages for files not found and removes the protection from all others.

/[NO]LOG

[Does not] list on the terminal the names of all the files unprotected by the command.

/NEWFILES

Removes protection from only those files having the current system date. The following example removes protection from the files created today:

```
.UNPROTECT/NEWFILES DU1:*.BAK
Files unprotected:
DU1:MERGE.BAK ? Y
```

/QUERY

Asks for a confirmation before removing protection from each file. In the following example protection is removed only from the file DU1:AAF.MAC:

```
.UNPROTECT/QUERY DU1:*.*
Files unprotected:
DU1:ABC.MAC   ? N
DU1:AAF.MAC   ? Y
DU1:MERGE.FOR ? N
```

/SETDATE[:date]

Assigns the specified date to all the newly unprotected files.

The following example removes protection from files and changes their dates to the current system date:

```
.UNPROTECT/SETDATE DU0:*.FOR
Files unprotected:
DU0:ABC.FOR
DU0:AAF.FOR
DU0:MERGE.FOR
```

/SINCE[:date]

Removes protection from only those files created on or after the specified date. The following command removes protection from all MAC files on DU0 created on or after April 21, 1991:

```
.UNPROTECT/SINCE:21:APR:91 DU0:*.MAC
```

/SYSTEM

Removes protection from system (SYS) files when you use wildcards in the file type. This example removes protection from all files on DU0 with the file name MM, including SYS files:

```
.UNPROTECT/SYSTEM DU0:MM.*
 Files unprotected:
DU0:MM.MAC
DU0:MM.OBJ
DU0:MM.SAV
DU0:MM.SYS
```

/WAIT

Waits for the volume to be mounted before executing the command. The following example removes protection from the file FILE.MAC on an RL02 disk:

```
.UNPROTECT/WAIT DL0:FILE.MAC
Mount input volume in DL0:; Continue? Y
DL0:FILE.MAC? Y
Mount system volume in DL0:; Continue? Y
```

V/VRUN

Runs a program as a completely virtual job. See also FRUN, R, RUN, and SRUN.

V program

or

VRUN program

The default device for V is SY, while the default device for VRUN is DK. In the following example, since V is a DCL command, RT–11 returns you to the keyboard monitor dot prompt when it finishes execution:

```
.V MACRO MYPROG MYPROG RET
.
```

Chapter 4

RT–11 Utilities

RT–11 utilities use the Command String Interpreter (CSI) to process the command lines typed to them.

CSI Command String Format

output-filespecs/options=input-filespecs/options

Syntax

output-filespec (up to three files)

dev:filnam.typ[n],...dev:filnam.typ[n]

input-filespec (up to six files)

dev:filnam.typ,...dev:filnam.typ

/option

/x:oval or /x:dval.

Mnemonic	Meaning
dev:	1- to 3-character device name
filnam	1- to 6-character file name
typ	0- to 3-character file type
[n]	length of output file; brackets are part of syntax
/x	option character
oval	optional octal argument or Radix–50 3-character argument
dval.	optional decimal argument; indicated by decimal point
=	separates output and input filespecs

Some utility programs, such as BINCOM, have particular meanings assigned to particular files in the CSI command string. In these cases, the syntax of the utility's command line is shown in detail.

See the *RT–11 System Utilities Manual* (Part I and Part II) for descriptions of BATCH, CONFIG, CONSOL, DATIME, EDIT, NITEST, RTMON, SPLIT, and VBGEXE. These utilities are not summarized in this manual.

BINCOM (Binary Compare Program)

```
.R BINCOM
*[listfile][,SIPPfile]=oldfile,newfile[/options]
```

/B	Compares bytes instead of words
/D	Compares two entire volumes
/E:n	Ends comparison at block *n*
/H	Displays help information on terminal
/O	Creates a differences listing file or SIPP command file even if no differences between the input files are found
/Q	Suppresses terminal output of differences
/S:n	Starts comparison at block *n*

BUP (Backup Utility Program)

/E/X	Restores .SYS files to SY when using wildcards
/F/X	Restores an entire saveset to one file
/G	Inhibits the bad-block scan on disk output volumes
/I	Backs up an entire volume to smaller volumes in image mode. Also used with /X during restore operations
/L	Displays a directory of a backup volume. /L/R displays a directory of a logical-disk file
/M	Inhibits rewinding magtape before appending next saveset to that magtape
/R	Creates logical-disk images of the files you want to back up. /R/X restores files from a logical disk
/S	Indicates the saveset containing a file you want to restore or the saveset from which you want to obtain directories
/V	verifies that input and output files match in a backup operation; /V/X verifies a restore operation; and /V:ONL/X verifies data in a previous backup operation (without restoring the data)
/W	Suppresses various informational messages
/X	Restores information backed up using BUP
/Y	Inhibits prompting for responses otherwise required from the terminal
/Z	Initializes a volume for use as an output volume in a backup operation
no option	Backs up files

DIR (Directory Listing Program)

/A	Lists directory alphabetically
/B	Includes starting block numbers in directory listing
/C:n	Lists directory in *n* columns; *n* can be 1 to 9
/D[:date]	Includes only files with *date*
/E	Lists entire directory, including unused spaces
/F	Lists short format directory in five columns
/G	Lists directory entry of specified file and all subsequent directory entries
/J[:date]	Lists files created on or after *date*
/K[:date]	Lists files created before *date*
/L	Lists volume directory in order of entry
/M	Lists unused areas
/N	Lists directory summary
/O	Gives sizes and block numbers in octal
/P	Lists all files except those you specify
/Q	Lists deleted files
/R	Sorts directory in reverse order; use with /S
/S[:xxx]	Sorts directory listing; *xxx* can be DAT, NAM, POS, SIZ, or TYP
/T	Lists only protected files
/U	Lists only unprotected files
/V[:ONL]	Includes volume ID and owner name as part of directory listing; with ONL, lists only ID and name

DUMP (File Dump Program)

/A	Displays the ASCII equivalent of each octal word or byte
/B	Outputs octal bytes
/E:n	Ends output at block *n*
/G	Ignores input errors
/N	Suppresses ASCII output
/O:n	Outputs only block *n*
/S:n	Starts output at block *n*
/T	Defines a magtape as non-RT–11 file structured
/W	Outputs octal words
/X	Outputs Radix–50 characters

DUP (Device Utility Program)

/B[:RET]	Writes FILE.BAD entries over bad blocks; use with /Z; with RET, retains FILE.BAD entries created by previous initialization
/C	Creates a file; use with /G:n
/D	Restores previously initialized volume
/E:n	Specifies last block number; use with /I or /K
/F	Prints names of files with bad blocks; use with /K
/G:n	Specifies starting block number; use with /C, /I, or /K
/H	Verifies after copying; use with /I
/I	Copies image of one volume to another
/J	Ignores a bad block on the input or output device and proceeds with the copy operation while displaying an error message.
/K	Scans a volume for bad blocks
/N:n	Defines number of directory segments; use with /Z; *n* can be 1 to 37(octal)
/O	Boots a volume or file
/Q	Boots a volume that is not RT V4 or later; use with /O
/R[:RET]	Scans volume for bad blocks and creates a block replacement table; with *RET*, retains previous table
/S	Consolidates free space on a volume
/T:n	Extends a file by *n* blocks; *n* free blocks must follow the file
/U[:dev]	Writes bootstrap into blocks 0, 2 through 5 of a volume; *dev* is the target device name
/V[:ONL]	Prints user ID and owner name; use with /Z to write new directory, ID, and name on volume; with ONL, writes only a new ID and owner name
/W	Waits for volume to be mounted before executing the command
/X	Prevents automatic reboot after using /S on system device
/Y	Suppresses query messages
/Z[:n]	Initializes device directory; *n* is the number of extra words in each directory entry

ERROUT (Error Log Report Writer)

/A	Displays report on each error and a summary report on all errors and I/O transfers
/F:date	Displays report of all errors that occurred on or after the date you specify
/S	Prints summary report only
/T:date	Displays report of all errors that occurred before the date you specify

FILEX (File Exchange Program)

/A	Transfers ASCII files character-by-character deleting nulls and rubouts; ^Z is interpreted as EOF
/D	Deletes a file; valid for DOS/BATCH, RSTS DECtape, interchange diskette
/F	Lists short directory
/I	Transfers in image mode
/L	Lists full directory
/P	Transfers in packed image mode
/S	Indicates volume is DOS/BATCH or RSTS format
/T	Indicates volume is DECsystem-10 DECtape
/U[:n]	Indicates volume is interchange diskette; *n* defines size of output record; default *n* is 80
/V[:ONL]	Writes a volume identification on interchange diskette; use with /U/Z; with ONL, writes only volume ID and keeps old directory
/W	Waits for volume to be mounted before executing the command
/Y	Suppresses query messages
/Z	Initializes device directory; valid for DOS/BATCH, RSTS DECtape, interchange diskette

FORMAT (Volume Formatting Program)

FORMAT can format RX02 and RX33 diskettes, and RK05, RK06, RK07, RD50, RD51, RD52, and RD53 disks. FORMAT can FORMAT/VERIFY:ONLY RX50, RL01 and RL02. RD5n disks can be formatted to run under DW: handler only.

/P[:n] Defines patterns to be used during verification pass. n pattern n pattern

```
 1 000000  100 021042
 2 111111  200104210
 4 163126  400 155555
10 125252 1000 145454
20 052525 2000 146314
40 007417
```

/S Formats diskettes in single-density format

/V[:ONL] Formats, then verifies volume; with ONL, verifies without formatting

/W Waits for volume to be mounted before executing the command

/Y Suppresses query messages

no option Formats without verification, assumes diskettes are double-density

IND (Indirect Command File Processor)

Options

/D	Deletes control file after processing
/N	Suppresses execution of KMON commands within a control file
/T	Lists each IND command line as it is executed
/Q	Suppresses display of keyboard commands and responses

Directives

Label Definition Directive

.label:	Assigns the name represented by *label* to a line in the control file so that the line can be referenced.

Symbol Definition Directives

.ASK [def:time] symbol prompt	Defines or redefines a logical symbol and assigns the symbol a logical (true or false) value based on a response to a prompt.
.ASKN [low:high:def:time] symbol prompt	Defines or redefines a numeric symbol and assigns it a numeric value based on a response to a prompt.
.ASKS [low:high:‘‘def’’:time] symbol prompt	Defines or redefines a string symbol and assigns it a string value based on a response to a prompt.
.DUMP symboltable	Displays local, global, and special symbol definitions.
.ERASE LOCAL [symbolname] .ERASE GLOBAL [symbolname]	Deletes local or global symbol definitions.
.PARSE str ‘‘ctr-str’’ sy1 sy2 ....syn	Breaks a string into substrings, which IND stores in string symbols.
.SETD numersymbol .SETO numersymbol	Redefines a numeric symbol to a decimal (.SETD) or octal (.SETO) radix.

Symbol Definition Directives

.SETL logicalsymbol logicalexpression	Defines or redefines a logical symbol and assigns it a logical value.
.SETN numersymbol numerexpression	Defines or redefines a numeric symbol and assigns it a numeric value.
.SETS stringsymbol stringexpression	Defines or redefines a string symbol and assigns it a string value.
.SETT logicalsymbol .SETT [mask] numersymbol .SETF logicalsymbol .SETF [mask] numersymbol	Defines or redefines a logical symbol or redefines bits within a numeric symbol, and assigns the symbol or bits a true or false value.
.VOL stringsymbol device	Assigns the volume identification of a volume to a string symbol.

File Access Directives

.CHAIN filespec[/options]	Closes the current control file, opens another control file, and resumes execution.
.CLOSE [#n] [filespec]	Closes an output data file.
.DATA [#n] text-string	Specifies a single line of data to be sent to an output data file.
.OPEN [#n] filespec	Creates an output data file. If the file you specify with .OPEN already exists, .OPEN creates a new tentative file with the same name. If you subsequently use the .CLOSE directive, .CLOSE will delete the old file and make the new file permanent. Use the .OPEN directive only when you wish to write to a file.
.OPENA [#n] filespec	Opens an existing file and appends subsequent data to it. If the file you specify does not already exist, .OPENA creates a new file. Use this directive only when you wish to write to a file.

IND (Indirect Command File Processor)

File Access Directives	
.OPENR [#n] filespec	Opens an existing file for use with the .READ directive. Use this directive only when you wish to read from a file.
.PURGE [#n]	Deactivates an output file, resulting in no changes to already existing files.
.READ [#n] stringsymbol	Reads the next record from a file into a string variable. The file must have been previously opened with .OPENR.
.STRUCTURE numsym dev	Determines the file structure of a specified file-structured device from boot block information.
.TESTFILE filespec	Determines whether or not a file exists.

Logical Control Directives	
.BEGIN	Marks the beginning of a Begin-End block.
.END	Marks the end of a Begin-End block.
.EXIT [value]	Terminates processing of either a Begin-End block or the current control file and returns control to the previous level; can also assign a value to the numeric symbol <EXSTAT>.
.GOSUB label	Branches to a subroutine within the control file.
.GOTO label	Branches to another location in the control file.
.ONERR label	Upon detecting an error, branches to another location in the control file.
.RETURN	Returns control from a subroutine to the line immediately following that subroutine's call.
.STOP	Terminates control file processing.

Logical Test Directives

.IF symbol operator expression action	Determines whether or not a symbol satisfies one of several possible conditions.
.IFDF symbol action .IFNDF symbol action	Determines whether or not a symbol is defined.
.IFENABLED opmode action .IFDISABLED opmode action	Determines whether an operating mode is enabled or disabled.
.IFLOA device action .IFNLOA device action	Determines whether or not a device handler has been loaded.
.IFT logicalsymbol action .IFT [mask] numersymbol action .IFF logicalsymbol action .IFF [mask] numersymbol action	Determines whether a logical symbol is true or false, or tests specific bits in a numeric symbol.
.TESTDEVICE devicename[:]	Returns information about a device in the string <EXSTRI>.
.TEST symbol .TEST stringsymbol matchstring	Tests a symbol or string.

Execution Control Directive

.DELAY time	Delays control file processing for a specified period of time.

Operating Mode Directives

.DISABLE mode1[,mode2,mode3...] .ENABLE mode1[,mode2,mode3...]	Disables or enables the operating modes, which are:

```
DATA        LOWERCASE   SUBSTITUTION
DCL         MCR         SUFFIX
DELETE      OCTAL       TIMEOUT
ESCAPE      PREFIX      TRACE
GLOBAL      QUIET
```

Increment/Decrement Numeric Symbols Directives	
.DEC numersymbol	Subtracts 1 from the value of a numeric symbol.
.INC numersymbol	Adds 1 to the value of a numeric symbol.

KED (Keypad Editor)

.KED
*[[out-filespec] [,jou-filespec]=] in-filespec [,jou-filespec] [,init-filespec]

DCL	CSI	Description
/ALLOCATE:size	/A:n	Allocates *n* blocks for output file
/**COMMAND**[:init-filespec]	**None**. Indicated by position in command line	Executes initialization file
/NOCOMMAND	/N	Does not execute initialization file
/CREATE	/C	Creates a new file
/INSPECT	/I	Inspects input file, does not allow changes
/JOURNAL[:outjou-filespec]	None. Indicated by position in command line	Produces a journal file
/NO**QUERY**	/Y	Executes your command without waiting for confirmation
/RECOVER[:injou-filespec]	None. Indicated by position in command line	Recovers a text file

LD.SYS (Logical-Disk Handler and Utility)

LD.SYS operates in two ways: either it can be used by programs as a handler to access logical disks, or it can be run as a program to initialize logical-disk parameters.

/A:ddd	Assigns logical name *ddd* to a logical-disk unit; use with /L:n
/C	Verifies all logical-disk assignments; must be alone on command line
/L:n	Assigns or deassigns logical-disk unit *n*
/R:n	Write locks logical-disk unit *n*
/W:n	Write enables logical-disk unit *n*

LET (LET Substitution Utility)

Options

Command /Option	Function
LET /HELP	Displays help summary
LET /LIST	Displays current character assignments
LET x/DELETE	Deletes the assignment for x
LET /DELETE	Deletes all assignments with the query *Are you sure?*
LET /DEL:ALL	Deletes all assignments without requesting confirmation

To enable LET, enter the command SET SL LET,KMON. This command assumes that SL is neither loaded nor on.

To define a substitution:

LET _x=string

where

_	tells SL that you are defining a symbol. This prevents SL from trying to substitute a string for the character that follows.
x	is a 1-character symbol that you want to equate with a string.
string	is a character string.

For example, LET _#=DX:MYPROG.MAC. The following LET substitution defines the F7 function key as the string *MACRO/LIST/CROSSREFERENCE*:

```
.LET F7=MACRO/LIST/CROSSREFERENCE RET
```

LIBR (Librarian)

.R LIBR
*[libraryfile][,listfile]=inputfiles[/options]

/A	Includes in library directory all global symbols including absolute global symbols
/C	Allows multiple input lines
/D	Deletes a module from a library file
/E	Extracts a module from a library and stores it as a .OBJ file
/G	Deletes a global symbol from a library directory
/M:n	Creates a macro library from an ASCII input file and allocates *n* blocks for the macro name directory
/N	Includes module names in library directory
/P	Includes psect names in library directory
/R	Replaces modules in a library file
/U	Updates (inserts and replaces) modules in a library file
/W	Produces a wide (132 column) library directory listing
/X	Creates a library with multiple global definitions
//	Allows multiple input lines until next occurrence of //
no option	Assumes module insertion

LINK (Linker)

.R LINK
*[binfile][,mapfile][,stbfile]=objfiles[/options]

/A	Lists global symbols in alphabetical order
/B:n	Sets bottom address of program to *n*; invalid with /H and /R
/C	Continues input on new line; do not use with //
/D	Allows duplicate library subroutines
/E:n:s	Extends program section in root segment to specific value
/F	Uses default FORTRAN library FORLIB.OBJ when linking
/G	Increases size of linker's library directory buffer
/H:n[:t]	Specifies highest address to be used by relocatable code in optional type DAS or IAS space; invalid with /B, /Q, /R, /Y
/I	Extracts specified global symbols from a library
/J	Generates an extended SAV image file which separates I- and D-space
/K:n	Inserts value of *n* into word 56 of block 0 as virtual .SETTOP high limit; *n* can be 1 to 32(decimal); valid only with /V
/L	Produces output file in LDA format; invalid with /R, /V
/M[:n]	Defines stack address
/N	Produces global cross-reference listing as part of load map
/O:n	Defines overlay structure; invalid with /L
/P:n	Changes amount of space linker uses for library routines list; default is 170
/Q	Specifies start addresses of up to eight root program sections; invalid with /R, /H
/R[:n]	Produces output in .REL format; *n* is the stack size; invalid with /B, /H, /K, /L, Q
/S	Allows maximum memory space for linker symbol table
/T[:n]	Defines transfer address
/U:n[:t]	Rounds up program-section size in root in optional type DAS or IAS space; *n* must be a power of 2
/V	Enables special XM monitor .SETTOP and .LIMIT features; invalid with /L

/V:na[nb]	Specifies an extended-memory overlay segment be mapped to virtual region *na* in optional partition *nb*
/W	Produces a wide load map listing (132 columns)
/X	Does not output bitmap if code is below 400
/Y[:n[:t]]	Starts a program section in the root on address boundary *n* in optional type DAS or IAS space; invalid with /H
/Z:n[:t]	Sets unused locations to *n* in optional type DAS or IAS space
//	Allows multiple input lines; use on first and last lines of input

MACRO (Macro Assembler)

.R MACRO
*[objfile][,listfile][,creffile]=inputfiles[/options]

Temporary files for MACRO and CREF are normally assigned to DK. Assign the logical name WF to reassign MACRO's work file; assign the logical name CF to reassign CREF's temporary file.

/C[:arg] Generates cross-reference table; *arg* can be

```
C - Control and program sections
E - Errors
M - Macro names
P - Permanent symbols
R - Registers
S - User-defined symbols
no argument - equivalent to /C:E:M:S
```

/D:arg Specifies .DSABL directives; *arg* can be

```
ABS - Produces absolute binary output
AMA - Assembles all relative addresses as absolute addresses
CDR - Treats source columns beyond column 72 as comments
DBG - Outputs ISD records as part of .OBJ file
FPT - Floating-point truncation
GBL - Treats undefined symbols as globals
LC  - Accepts lowercase ASCII input
LSB - Local symbol block
MCL - Automatic .MCALL
PNC - Binary output
REG - Defines default register mnemonics
```

/E:arg Specifies .ENABL directives; see values for *arg* under /D

/L:arg Enables listing directives; *arg* can be

```
BEX - Binary expansions of ASCII text
BIN - Generated binary code
CND - Unsatisfied conditionals and all (.IF and .ENDC
      statements)
COM - Comments
LD  - Listing directives having no arguments
LOC - Location counter
MC  - Macro calls and repeat range expansions
MD  - Macro definitions and repeat range expansions
ME  - Macro expansions
MEB - Macro expansion binary code
SEQ - Line numbers
SRC - Source code
SYM - Symbol table
TOC - Table of contents
TTM - Listing output format in 80-column format;
      default is 132-column format
```

/M Macro library input file

/N:arg Disables listing directives; refer to values for *arg* under /L

/P:n Specifies pass 1 or pass 2 assembly of input source file

ODT/VDT (Octal Debugger)

Commands

RET	Closes open location and accepts next command
LF	Closes current location and opens next sequential location
^	Opens previous location
_	Indexes contents of opened location by PC and opens resulting location
>	Uses contents of opened location as relative branch and opens that location
<	Returns to sequence prior to @, >, or _ and opens next location
@	Uses contents of opened location as absolute address and opens that location
r/	Opens word at location *r*
/	Reopens last opened location
r\	Opens byte at location *r*
\	Reopens last opened location as byte
I	Displays address of opened location relative to the closest relocation register
nI	Displays address of opened location relative to relocation register *n*
$n/	Opens general register *n*
$B/	Opens first word of breakpoint table
$C/	Opens constant register
$F/	Opens format register
$M/	Opens first mask register
$P/	Opens priority register
$R/	Opens first relocation register
$S/	Opens status register
r;nA	Displays *n* bytes of ASCII starting at location *r*, then allows *n* bytes of input from the terminal
;B	Removes all breakpoints
r;B	Sets next free breakpoint at location *r*
r;nB	Sets breakpoint *n* at location *r*
;nB	Removes breakpoint *n*

n;C	Stores *n* in constant register
r;E	Searches for instructions that reference effective address *r*
;F	Fills memory with the contents of the constant register
r;G	Starts execution of program at location *r*
;I	Fills memory bytes with low 8 bits of the constant register
r;O	Calculates offset from current location to location *r*
;P	Execution proceeds from breakpoint
n;P	Execution proceeds from breakpoint; stops after encountering the breakpoint *n* times
;R	Sets all relocation registers to 177777
;nR	Sets relocation register *n* to 177777
m;nR	Sets relocation register *n* to value *m*; default *n* is 0
R	Subtracts contents of closest relocation register less than or equal to the contents of the current location from the contents of the location, and displays result
nR	Subtracts contents of relocation register *n* from contents of current location and prints result
!	Calculates address of the current location relative to the closest relocation register less than or equal to the address of the current location, and displays result
n!	Calculates address of the current location relative to relocation register *n*, and displays result
;S	Disables single-step mode
;nS	Enables single-step mode; *n* can be any integer
n;W	Searches for words matching value *n*
X	Displays contents of opened location as Radix–50; accepts three Radix–50 characters as input

PAT (Object Module Patch Program)

/C[:n]	Calculates [or verifies] module checksum

PIP (Peripheral Interchange Program)

/A	Copies files in ASCII mode
/B	Copies files in formatted binary mode
/C[:date]	Includes only files with *date*
/D	Deletes files
/E	Waits for volume to be mounted before executing the command
/F	Marks output files as protected
/G	Ignores any input errors
/H	Verifies output file; do not use with /A or /B
/I[:date]	Copies only files created on or after *date*
/J[:date]	Copies files created before *date*
/K:n	Transfers *n* copies of the output files to a sequential device, such as LP:, PC:, or TT:
/M:n	Controls positioning and rewinding of magtape; See the next section
/N	Does not copy or rename a file if a file of the same name exists on the output device
/O	Deletes a file on the output device before copying, if a file of a given name already exists
/P	Copies or deletes all files except those specified
/Q	Queries before performing operation
/R	Renames input filename to output filename
/S	Copies files one block at a time
/T[:date]	Puts *date* on all files copied or renamed
/U	Copies and concatenates specified files
/V	Copies files from one large volume to several small volumes
/W	Logs operations on console
/X	Displays informational rather than fatal messages for files not found, and processes all others

/Y	Includes .SYS files in operation
/Z	Removes protection from output files
no option	Copies files in Image mode

Options /D, /F, /K, N, /O, /R, /T, /Z are invalid or restricted with magtape.

PIP—Use of /M:n

Effect when reading from magtapes

n = 0	The magtape rewinds and PIP searches for the file. If more than one name is given, PIP rewinds for each file. If a wildcard is used, PIP rewinds once and then copies all matching files as they are encountered.
n = +n	PIP goes to file with sequence number *n*. If the file at that position matches specified name, the file is copied; otherwise PIP returns *file not found* error. If a wildcard is used, PIP begins the search at file *n*.
n = -1	PIP begins the search at current position without rewinding.

Effect when writing to magtapes

n = 0	The tape rewinds before each file is copied and PIP searches for a duplicate file name. If a file of the same name is found, PIP prints a warning and does not copy the file. Otherwise, the file is written at LEOT.
n = +n	PIP goes to file sequence number *n* and enters specified file. If LEOT is encountered before file *n*, an error is printed. If more than one file name is specified or a wildcard is used, the tape does not rewind and PIP does not check for duplicate names.
n = -1	PIP goes to LEOT and enters specified file without checking for duplicate names.
n = -2	The tape rewinds before each copy operation. PIP enters the file at LEOT or at the occurrence of a duplicate file name.

QUEMAN (Queue Manager)

/A	Halts QUEUE
/C[:date]	Prints only files created on *date*
/D	Deletes files after printing
/H:n	Prints *n* banner pages
/I[:date]	Prints only files created on or after *date*
/J[:date]	Prints files created before *date*
/K:n	Prints *n* copies of a file
/L	Lists current contents of the queue
/M	Deletes a file from the queue
/P	Sets QUEUE defaults for /H and deletion of work file
/Q	Asks for confirmation before printing
/R	Resumes or restarts current job
/S	Suspends current job; resume with /R
/W	Logs all files printed
/X	Prints error messages for files not found and processes all others
//	Allows multiple input command lines

RESORC (Resource Program)

/A	Combines all options except /Z
/C	Identifies system device and lists monitor SET options in effect
[dd:]/D	Lists available device handlers and their vectors; with *dd:*, lists information about only that device
/H	Lists system hardware configuration
/J	Lists currently loaded jobs
/L	Lists device assignments
/M	Lists monitor type, version number, and patch level
/O	Lists special features included by SYSGEN
/Q	Lists contents of SPOOL and/or QUEUE queue
/S	Lists disk subsetting in effect
/T	Lists status and options in effect for active terminals on a multiterminal system
/V	Lists release and version numbers for RT–11 image modules
/X	Lists organization of physical memory
/Z	Combines /C, /H, /J, /M, /O

SIPP (Save Image Patch Program)

```
.R SIPP
*[commandfile]=inputfile[/options]
```

Options

/A	Prevents automatic update of location 50, window definition block, overlay handler, or overlay tables
/C	Asks for checksum at completion of the patch
/D	Prints checksum at completion of the patch
/L	Does not modify the input file

Commands

RET	Closes current location, opens and displays next location
LF	Closes current location, opens and displays next location
n RET	Deposits value *n* in current location, closes current location, opens and displays next location
/	Advances in word mode
/ RET	Reopens current location as a word
\ RET	Reopens current location as a byte
\	Advances in byte mode
^ RET	Closes current location, opens and displays previous location
n^ RET	Deposits value *n* in current location, closes current location, opens and displays previous location
;A	Interprets location contents as ASCII
;O	Gives values in octal, accepts input in octal
;Ax	Deposits ASCII character *x* in low byte of current location, opens and displays high byte in octal, accepts further input in ASCII
;R	Interprets location contents as Radix–50 characters
;Ryyy	Deposits Radix–50 characters *yyy* in current location, closes current location, opens next location and gives value in octal, accepts further Radix–50 character input
;S	Prompts for value for which to search
;V	Prints all previously entered modifications

CTRL/Y	Prompts for checksum, if /C used, and installs patch
CTRL/Z	Backs up to previous SIPP prompt

SLP (Source Language Patch Program)

.R
*[patchedfile][,listfile]=inputfile[,commandfile][/options]

Options

/A	Disables audit trail generation
/B	Uses spaces instead of tabs to position audit trail text
/C	Computes checksum for file
/C:n	Validates checksum for file
/D	Creates a double-spaced listing
/L:n	Defines maximum length of a source line, in characters
/N	Suppresses creation of backup file
/P:n	Starts audit trail in column *n*
/S:n	Defines length of audit trail; maximum value for *n* is 16(decimal)
/T	Retains trailing tabs and blanks in updated file

Command File Operators

_	Indicates start of an update
\	Disables audit trail
%	Enables audit trail
/	Indicates the end of an update or series of updates
//	Separates concatenated SLP command files
<	Indicates that the next character should be taken as source input, not interpreted as a command; use in front of _, \, %, /, and < when they must be inserted in a file

SRCCOM (Source Compare Program)

.R SRCCOM
*[listfile][,SLPfile]=oldfile,newfile[/options]

/A	Specifies audit trail for SLP command file
/B	Compares blank lines
/C	Ignores comments and spacing
/D	Copies newfile to listfile and inserts change bars and bullets at left margin of listfile to mark differences
/F	Includes formfeeds in output file
/L:n	Defines number of lines that must agree to constitute a match; *n* can be 1 to 310; default is 3
/S	Ignores spaces and tabs when comparing
/T	Includes trailing spaces and tabs in comparison
/V:i:d	Defines characters to use as markers for inserts and deletions in place of change bars and bullets; use with /D; *i* and *d* are the numeric values for the ASCII characters to use.

TRANSF (RT–11 Native Transfer Utility)

TRANSF outfile[/options]=infile[/options]

where:

outfile	specifies the file to which you want a file copied.
infile	specifies the file you want to copy.
options	specifies the options.

Options

/S	Enables log messages and rings terminal bell as files are transferred.
/T	Specifies which file is the RT–11 stand-alone system file.
/W	Displays log messages during file transfers, but does not ring terminal bell.

TRANSFER (RSX/VMS Native Transfer Utility)

TRANSFER infile/qualifier[s] [outfile/qualifier[s]]

where:

infile	specifies the file you want to copy
outfile	specifies the file to which you want to transfer the input file
/qualifier	specifies a TRANSFER qualifier

Command Qualifiers	**File Qualifiers (input or output)**
{ /HELP /PROMPT /VERSION }	{ /ASCII[:n] /BINARY[:n] /FORTRAN[:n] /IMAGE[:n] } /LOG /PROGRESS[:n] { /REMOTE /TERMINAL } /STATISTICS

{ } specifies mutually exclusive qualifiers.

TRANSFER (RSX/VMS Native Transfer Utility)

Qualifier	Type	Function
/ASCII[:n]	mode	formats the output as ASCII.
/BINARY[:n]	mode	formats the output as a binary OBJ file.
/FORTRAN[:n]	mode	transfers files containing FORTRAN carriage-control characters.
/HELP	control	displays information about TRANSFER.
/IMAGE[:n]	mode	transfers files without performing any record translations on them.
/LOG	control	creates a log of the names of all the files transferred.
/PROGRESS[:n]	control	displays the progress of the transfer at specific intervals while the operation is taking place.
/PROMPT	control	operates in an interactive mode that displays questions, indicates defaults, and accepts input.
/REMOTE	control	specifies the file on the local RT–11 system.
/TERMINAL	control	specifies the file on the local RT–11 system (this qualifier is identical to /REMOTE).
/SPOOL	control	queues the TRANSFER output file to the default VMS printer queue.
/STATISTICS	control	displays the number of retries and the number of characters saved through compression encoding.
/VERSION	control	displays the TRANSFER utility's version number.

VTCOM (Virtual Terminal Communications)

VTCOM enables you to connect your stand-alone RT–11 operating system to another computer's operating system.

To issue a VTCOM command, first enter terminal mode by typing ^ CTRL/F or CTRL/X and the system-job name. Then, at the VTCOM> prompt, type ^ CTRL/P.

Commands

Command	Function
^x	Lets VTCOM transmit CTRL characters that would normally be intercepted: ^ CTRL/B, CTRL/F, CTRL/O, CTRL/P, CTRL/Q, CTRL/S.
BREAK	Transmits a break signal to host.
CLEAR	Clears any ^ CTRL/S characters that have been sent, and starts sending characters to the terminal again.
CLOSELOG	Stops recording input in a log file and closes the log file.
CONTINUE	Returns your system to terminal mode.
CTRL/P	Sends a ^ CTRL/P character to the host, VTCOM normally intercepts ^ CTRL/P characters and interprets them as a request to enter a VTCOM command.
DIAL	Causes the modem to dial the telephone dial string you specify.
EXIT	Terminates the VTCOM program and the XC or XL handler and closes any open logging file. To restart VTCOM, you must use the FRUN or SRUN command.
FAST	Lets VTCOM transmit ASCII characters to the host at high speed during a SEND operation.
HANGUP	Breaks the modem connection.
HELP	Displays a list of VTCOM commands.
LOG	Resumes recording data in a log file after a NOLOG command.
NOLOG	Suspends the recording of data in a log file.
OPENLOG	Opens a log file to receive ASCII input from the host system, and starts recording input in the log file.

VTCOM (Virtual Terminal Communications)

Command	Function
PAU SE	Ends VTCOM program control.
RE SET	Halts file transfers using TRANSF and VTCOM SEND operations.
SEL ECT	Lets you specify a port on a Mini-Exchange.
SEN D	Transfers an ASCII file from your stand-alone system to a host as if the file were being typed.
SH OW	Displays the following VTCOM characteristics: Data transfers in progress Logging status—on or off SEND status—slow or fast Current dial string
SL OW	Causes VTCOM to transmit ASCII characters to the host at slow speed during a SEND operation.