RSX-11M
Command Language Manual
Order No. AA-L672C-TC

digital
software

RSX–11M
Command Language Manual

Order No. AA–L672C–TC

RSX–11M Version 4.2

The following are trademarks of Digital Equipment Corporation:

DEC	DIBOL	PDT
DEC/CMS	EduSystem	RSTS
DEC/MMS	IAS	RSX
DECnet	MASSBUS	UNIBUS
DECsystem-10	MicroPDP-11	VAX
DECSYSTEM-20	Micro/RSTS	VMS
DECUS	Micro/RSX	VT
DECwriter	PDP	digital

ZK2868

CONTENTS

CONTENTS

CONTENTS

CONTENTS

PREFACE

MANUAL OBJECTIVES

The RSX-11M Command Language Manual provides information needed to use a terminal to do work on an RSX-11M system. Important system concepts are introduced and explained in the context of the DIGITAL Command Language (DCL). DCL is a user-oriented command language with English words and prompting for required command elements.

INTENDED AUDIENCE

This manual is intended for any user of an RSX-11M system. The manual is a reference manual with many tutorial elements, but new users should read the Introduction to RSX-11M first.

A user is anyone who interfaces with the system to perform some task.

There are two kinds of users: privileged and nonprivileged. Privileged users have access to functions that control and modify system operation as well as functions that permit program development and system maintenance. Nonprivileged users have access to functions required for program development and local maintenance only.

STRUCTURE OF THIS MANUAL

The manual is organized around the major operating system functions. Following the introduction, each major function has a chapter to itself. All commands relating to the function are described in the chapter. The last chapter is an annotated alphabetical listing of all DCL commands.

Chapter 1 introduces the basic concepts of DCL, explains how DCL is implemented on RSX-11M, and includes many hints on using DCL. The chapter also includes summaries of the remainder of the manual, as well as general information about DCL, the operating system, and various utilities and system tasks.

Chapter 2 lists the most common forms of SET and SHOW commands.

Chapter 3 covers terminal operations, including logging in and logging out, keyboard terminology and functions, setting and displaying terminal attributes, broadcasting messages, and using the system HELP commands.

Chapter 4 introduces file-handling operations on RSX-11M, including commands to create and delete files, list directories, and rename, copy, type, and print files. Wildcards and defaults are explained.

Chapter 5 covers peripheral devices and introduces the system I/O terminology. The chapter explains the relationships between hardware devices and the software file system, and describes how to prepare scratch disks and magnetic tapes for use on the system. Commands affecting devices and software volumes are described.

Chapter 6 introduces the program-development facilities of the system and the DCL commands used with these facilities.

Chapter 7 explains how tasks run in the system and how they are named, installed, fixed in memory, and controlled while running. In addition, the chapter describes commands to abort tasks, to place tasks in the clock queue, and to display information about tasks in the system.

Chapter 8 covers the DCL system-control commands.

Chapter 9 is the alphabetical listing of all DCL commands and their formats, including comments to remind you of specific points about many commands and command elements. Each entry in Chapter 9 includes a cross-reference to the full command description in an earlier chapter.

Appendix A explains error messages common to several DCL commands. Other command-specific error messages are explained in the full command descriptions.

Appendix B lists the relationships between DCL and the rest of the operating system.

ASSOCIATED MANUALS

Other documents related to RSX-11M are described in the RSX-11M/RSX-11S Information Directory and Index.

If you are entirely new to computers or to DCL, you should read the Introduction to RSX-11M and follow the instructions there.

Although this manual is meant to be self-contained, you will find important information in the RSX-11M/M-PLUS MCR Operations Manual and the RSX-11M/M-PLUS Utilities Manual.

Programmers should see the RSX-11M/M-PLUS Guide to Program Development.

CONVENTIONS USED IN THIS MANUAL

A number of conventions are used in the command descriptions in this manual:

Convention	Meaning
CTRL/a	A symbol that indicates the CTRL key; it must be held down while another key is pressed. For example, CTRL/Z means hold down the CTRL key while pressing Z. In examples, this control key sequence is shown as ^a; for example, ^O indicates the result of CTRL/O because that is how the system echoes most control key combinations.
XXX	A 1- to 3-character key symbol. For example, RET indicates the RETURN key, LF indicates the LINE FEED key, and ESC indicates the ESCAPE key.

.
.
.

The vertical ellipsis shows where elements of command input have been omitted because they are irrelevant to the point being discussed.

[/qualifier]

Any command field enclosed in brackets is optional. If the brackets include syntactical elements, such as dots (.) or slashes (/), those elements are required for the field. If the field appears in lowercase, you are to substitute a legal command element if you include the field.

[g,m]

This signifies either a User Identification Code (UIC) or a directory. The g is a group number and the m is a member number. Where a UIC or directory is required, only one set of brackets is shown, as [g,m]. Where the UIC or directory is optional, two sets of brackets are shown, as [[g,m]]. The UIC identifies a user and is used mainly for controlling access to files and privileged system functions.

UPPERCASE

Any command field in uppercase indicates the legal form of the command. If you type it in that form, it will work as described. Most DCL commands have abbreviations.

lowercase

Any command field in lowercase is to be substituted for. Usually the lowercase word identifies the kind of substitution expected, such as filespec, which indicates that you should fill in the file specification.

/qualifier

Any command element preceded by a slash (/) is a DCL qualifier. Command qualifiers alter the action of a command they are attached to. Parameter qualifiers modify the action of the command as it affects that parameter.

parameter

Required command fields are generally called parameters. The most common parameters are file specifications. Parameters are preceded by blanks or DCL prompts.

:argument

Some parameters and qualifiers can be altered by the inclusion of arguments preceded by a colon. An argument can be either numerical (COPIES:3) or alphabetical (NAME:KAT). The equals sign (=) can be substituted for the colon to introduce arguments. COPIES=3 and COPIES:3 are the same.

filespec

A full file specification includes device, directory, file name, file type, and version number, as in this example:

    DB1:[303,5]HIPPITY.HOP;2

Full file specifications are rarely needed. If you do not give a version number, the highest numbered version will be used. If you do not give a directory, the default directory will be used. Some system functions default to particular file types. See Chapter 4 for more information on file specifications. See also the individual command descriptions.

Some commands accept a file specification with a DECnet node name. See Chapter 1 for more information.

{A B}   A number of options between braces and separated by vertical bars means that you are to choose one of the options listed.

See Chapter 1 for more information on DCL conventions.

SUMMARY OF TECHNICAL CHANGES

The RSX-11M Command Language Manual has been revised to reflect
changes made to the software for Version 4.2. Not all of the
alterations made to the software for this version of RSX-11M resulted
in changes to this manual; see the RSX-11M Release Notes for a
comprehensive list of all software changes.

The following is a list of DCL commands and qualifiers new to this
version of RSX-11M:

ANALYZE/MEDIA[/qualifier[s]] ddnn:
              /ALLOCATE
              /BAD_BLOCKS
              /BAD_BLOCKS/EXERCISE:(n,m)
              /BAD_BLOCKS/NOEXERCISE
              /[NO]EXERCISE[:(n,m)]
              /OVERRIDE
              /RETRY
              /SHOW

   New command. The ANALYZE/MEDIA command determines if bad blocks
   exist on a disk volume and records their locations for use by the
   backup and restore utilities and the INITIALIZE command.

APPEND[/qualifier[s]]
       /NOWARNINGS
       /REWIND
       /SHARED

   New qualifiers.

BACKUP[/qualifier[s]]
       /CREATED:arg
               AFTER:(dd-mmm-yy hh:mm)
               BEFORE:(dd-mmm-yy hh:mm)
       /APPEND

   New qualifiers.

COPY[/qualifier[s]]
     /ALLOCATION:n
     /NOWARNINGS
     /OVERLAY
     /PRESERVE_DATE
     /REWIND
     /SHARED

   New qualifiers.

CREATE/DIRECTORY[/qualifier]
                    /NOWARNINGS
                    /OWNER_UIC

   New qualifiers.

DELETE[/qualifier[s]]
          /[NO]CONFIRM
          /NOWARNINGS

   New qualifiers.

DIRECTORY[/qualifier[s]]
              /NOWARNINGS
              /REWIND

   New qualifiers.

DISMOUNT[/qualifier[s]]
            /SAVE
            /TERMINAL
            /[NO]UNLOAD

   New qualifiers.

INITIALIZE[/qualifier[s]]
              /BAD_BLOCKS:arg
                              AUTOMATIC
                              (AUTOMATIC,MANUAL)
                              MANUAL
                              NOAUTOMATIC
                              OVERRIDE
                              (OVERRIDE,MANUAL)
              /DENSITY:arg
                          HIGH
                          LOW
              /LABEL:VOLUME_ACCESSIBILITY:"c"

   New qualifiers.

INITIALIZE/UPDATE[/qualifier[s]]
                      /DENSITY:arg
                                  HIGH
                                  LOW

New arguments.

INSTALL[/qualifier[s]]
          /[NO]INTERPRETER

   New qualifier.

LINK[/qualifier[s]]
        /[NO]PRINT

   New qualifier.

```
MOUNT[/qualifier[s]]
      /PROCESSOR:arg
            UNIQUE
```

Changed argument to existing qualifier.


```
PRINT[/qualifier[s]]
      /AFTER:TOMORROW
      /NOWARNINGS
```

New qualifiers.


```
PURGE[/qualifier[s]]
      /NOWARNINGS
```

New qualifier.


```
RENAME[/qualifier[s]]
        /NEW_VERSION
        /NOWARNINGS
```

New qualifiers.


```
RUN[/qualifier[s]]
    /[NO]IO_PAGE
    /[NO]SLAVE
```

New qualifiers.


```
SET DEVICE:ddnn:/qualifier[s]
              /[NO]CHECKPOINT_FILE[:n]
```

New qualifier.


```
SET FILE[/qualifier[s]] filespec[s]
        /END_OF_FILE:(BLOCK:n,BYTE:n)
        /ENTER:synonym_filespec
        /NOWARNINGS
        /REMOVE
        /REWIND
        /TRUNCATE
```

New command. SET FILE establishes certain file attributes.


```
SET HOST nodename
```

New command. SET HOST connects your terminal to a remote system.
You issue this command after you have logged in to your current
system. Both your current and remote system must run DECnet
software.


```
SET SYSTEM/qualifier
        /NETWORK_UIC:[g,m]
        /POOL/LIMITS
```

New qualifiers.

```
SET TERMINAL[:ttnn:]/qualifier[s]
                /INQUIRE
                /[NO]ADVANCED_VIDEO
                /[NO]ANSI_CRT
                /[NO]BLOCK_MODE
                /DEC_CRT
                /DTC01
                /EDIT_MODE
                /HOSTSYNC
                /LA50
                /LA210
                /LN03
                /LQP02
                /LQP03
                /[NO]PARITY[:type]
                            ODD
                            EVEN
                /[NO]PASTHRU
                /PRINTER_PORT
                /PRO_SERIES
                /[NO]REGIS
                /[NO]SOFT_CHARACTERS
                /[NO]TTSYNC
                /VT200_SERIES
```

   New qualifiers.

SHOW HOST

   New command. SHOW HOST displays the name of the processor to which
   your terminal currently is connected. It also shows you the name
   and version number of the operating system running on the
   processor.


```
SHOW SYSTEM[/qualifier]
          /NETWORK_UIC
          /POOL/LIMITS
```

   New qualifiers.


```
SHOW TASKS[:taskname]/qualifier[s]
                /DEVICE:ddnn:
```

   New qualifier.


```
SHOW TERMINAL[:ttnn:][/qualifier]
                /[NO]ADVANCED_VIDEO
                /[NO]ANSI_CRT
                /[NO]BLOCK_MODE
                /[NO]DEC_CRT
                /DTC01
                /[NO]EDIT_MODE
                /HOSTSYNC
                /[NO]LA50
                /[NO]LA210
                /LN03
                /LQP02
                /LQP03
                /[NO]PARITY
                /[NO]PASTHRU
```

```
/PRINTER_PORT
/PRO_SERIES
/[NO]REGIS
/[NO]TTSYNC
/[NO]VT200_SERIES
```

New qualifiers.


```
TYPE[/qualifier[s]
    /NOWARNINGS
    /SHARED
```

New qualifiers.


```
UNLOCK[/qualifier[s]]
       /NOWARNINGS
```

New qualifier.

The following is a list of new devices supported by RSX-11M Version 4.2:

DHU11	RQDX2
DEUNA	RQDX3
KDA50	RUX50
RD52	TK25
RD53	TK50

The following is a list of new terminal types supported by RSX-11M Version 4.2:

DTC01	LQP02
LA50	LQP03
LA210	PRO-series terminals
LN03	VT200-series terminals

CHAPTER 1

HOW TO USE DCL

DCL is the DIGITAL Command Language. DCL provides RSX-11M users with
an extensive set of commands for interactive program development,
device and file manipulation, and interactive program execution and
control.

If you are new to computers, or if you wish a quick review of DCL and
the operating system, you should read the Introduction to RSX-11M.
This manual includes a guided tour of the operating system, including
an interactive terminal session that introduces all of the most
commonly used DCL commands.

DCL commands are full words that describe the action to be taken, not
abbreviations or mnemonics.   Thus, if you wish to set a terminal to
lowercase, you type SET TERMINAL/LOWERCASE. Because DCL commands and
command elements are full words, they are self-documenting.

You are not required to use the full form of DCL commands at the
terminal, however.   Usually, you need type only the command elements
required to form a unique command. For instance, SE TERM/LOW will
also set a terminal to lowercase, but SE TERM/LO will not work because
there is also a command SET TERMINAL/LOCAL. See Section 1.2 for more
information on how DCL commands are parsed.

1.1 RSX-11M AND DCL: BASIC CONCEPTS

RSX-11M systems may have one or more command line interpreters (CLIs).
All systems include MCR (the Monitor Console Routine). Many systems
also include DCL, and some systems include other CLIs. Both MCR and
DCL include commands to invoke most system tasks and utilities and to
set and display certain system characteristics.   In general, MCR
commands name tasks, such as PIP, a utility used to manipulate files
(for example, to copy or type them), while DCL commands specify
actions directly, as in the COPY command or the TYPE command.

MCR is the fundamental command line interpreter (CLI) for the RSX-11M
operating systems.   MCR also provides commands to set and display
certain system characteristics.   MCR provides the most   direct
interface with the RSX-11M operating system.

In general, MCR commands must be entered in exact syntax.   MCR
commands follow no set syntax rules, however. Most MCR commands are
abbreviations or mnemonics.

DCL is an optional user-oriented CLI included in most systems with
many users.   Commands in DCL are English-like words and follow
well-defined syntax rules. Full commands are self-documenting.   DCL
is designed for consistency and ease of use.

DCL is the command language used on many of DIGITAL operating systems. In particular, RSX-11M DCL is designed for compatibility with future releases of VAX/VMS DCL. See Section 1.7 for more information on DCL compatibility. DCL on RSX-11M systems is a CLI task that translates DCL commands into MCR commands for execution by the system. The DCL command SET DEBUG command displays the MCR translation for any DCL command on your terminal. See Section 1.2.15 for more information on the SET DEBUG command.

Depending on the kind of use you make of your system and the nature of your system, you may find it more convenient to use one CLI or the other, or both. All nonprivileged system functions are available directly from DCL, but some privileged functions are not. All program-development facilities and all common utility functions are available from DCL.

Because RSX-11M is designed to be tailored to the needs of each installation, not every feature of DCL described in this manual is available on every system. Some DCL commands depend on layered products that may not be available at your installation. Many features are system-generation options that may not have been selected at the time your system was generated. You should see your system manager or another knowledgeable system user if some feature described in this manual does not appear to be available at your installation.

1.2 THE DCL COMMAND LINE

A command consists of a command name or verb describing the action the system is to take. Most commands also include one or more parameters and qualifiers to further define the action of the command. Qualifiers are preceded by a slash (/) and parameters are preceded by a space. Both qualifiers and parameters can take arguments. Arguments are preceded by a colon (:).

Commands are passed to the operating system with a terminator, either RETURN or ESC. (The ESC key may be labeled SEL or ALT MODE. Its action differs from that of the RETURN key in that the cursor does not return to the left margin and no monitor prompt is returned after execution is completed.)

Unless the action of the RETURN greatly differs from what is expected, the examples and command formats in this manual do not include an indication that each line in the command is terminated by a carriage return. (See Section 1.2.13, for a discussion of command lines too long to fit on one line of your terminal.)

Some commands require parameters or arguments as part of the command line. If you fail to supply a required command element, DCL prompts you with one or two words indicating the general nature of the required element. If you do not understand the prompt, type a question mark (?) for help. (In some cases, an omission causes an error rather than a prompt.)

Section 1.2.1 discusses prompting in detail.

This section introduces the rules governing the use of DCL. The examples in this section are intended to illustrate these rules, not to illustrate the full capabilities of the command. For more detail, see the individual command descriptions.

1.2.1 Prompting

The prompts teach you the form of a command by requesting that you supply required command elements.

For example, the RENAME command works as follows:

```
    $ RENAME  (RET)
    From?  BROWNS.STL  (RET)
    To?   ORIOLES.BLT  (RET)
```

The one-line format for RENAME is as follows:

```
    $ RENAME BROWNS.STL ORIOLES.BLT  (RET)
```

The formats can be mixed. DCL prompts for whatever you leave out. For example:

```
    $ RENAME BROWNS.STL  (RET)
    To?   ORIOLES.BLT  (RET)
```

There are no defaults for prompts. You must supply a response to any prompt. If you do not wish to continue with the command, press CTRL/Z.

1.2.2 Qualifiers

Qualifiers modify the action of the command. Qualifiers always start with a slash (/) and are generally optional.

Qualifiers are either command qualifiers or parameter qualifiers. Most qualifiers are command qualifiers. In this manual, command qualifiers are always shown as modifying the command verb, as in this example:

```
    $ TYPE/TODAY *.HLP  (RET)
```

However, most command qualifiers can appear anywhere in the command line. Another name for these qualifiers is floating qualifiers. The following examples illustrate how command qualifiers can float:

```
    $ TYPE *.HLP/TODAY  (RET)

      or

    $ TYPE  (RET)
    File(s)?  *.HLP/TODAY  (RET)

      or

    $ TYPE  (RET)
    File(s)?  /TODAY  (RET)
    File(s)?  *.HLP  (RET)
```

You can mix formats and get exactly the same results, as in the following example:

```
    $ TYPE/TODAY  (RET)
    File(s)?  *.HLP/EXCLUDE:HELPF.HLP;*  (RET)

      or
```

```
$ TYPE/TODAY (RET)
File(s)? *.HLP/ (RET)
Qualifier? EXCLUDE:HELPF.HLP;* (RET)
```

    or

```
$ TYPE/TODAY (RET)
File(s)? / (RET)
Qualifier? EXCLUDE:HELPF.HLP;* (RET)
File(s)? *.HLP (RET)
```

Note that you are prompted for a qualifier when a slash with no qualifier attached appears on the command line. When you supply the qualifier, do not type the slash again.

Almost all command qualifiers can float in this way. In a few cases, the command qualifier must appear directly after the command it modifies. Whenever two similar commands are described separately, such as ASSIGN and ASSIGN/REDIRECT or CREATE and CREATE/DIRECTORY, the distinguishing qualifier cannot float. Other qualifiers to such commands can still float, however, as in this example:

```
$ CREATE/DIRECTORY (RET)
Device, UFD? [303,5]/ALLOCATION:5 (RET)
```

which is the same as the following example:

```
$ CREATE/DIRECTORY/ALLOCATION:5 [303,5] (RET)
```

Parameter qualifiers, sometimes called file specification qualifiers, cannot float. Usually a parameter qualifier must be attached to a file specification, because most DCL parameters are file specifications.

Many qualifiers can be negated by prefixing NO or - (minus) to the qualifier name. Thus, the following command:

```
$ MACRO/OBJECT SIMPLE.MAC (RET)
```

directs the MACRO-11 Assembler to make an object file, while the following command:

```
$ MACRO/NOOBJECT SIMPLE.MAC (RET)
```

    or

```
$ MACRO/-OBJECT SIMPLE.MAC (RET)
```

directs the MACRO-11 Assembler to omit the object file. In the first example, /OBJECT is the default qualifier and need not be included explicitly. In other words, unless your MACRO command includes the /NOOBJECT qualifier, an object file is produced.

1.2.3   HELP

HELP is available from the system for all DCL commands and also for many other aspects of the system. You can get help through the HELP command or by typing a question mark (?) in response to any DCL prompt.

For instance, if you wish help on the TYPE command, type the following:

```
$ HELP TYPE (RET)
 TYPE[/qualifier[s]] filespec[s]
        /DATE:dd-mmm-yy
        /SINCE:dd-mmm-yy
        /THROUGH:dd-mmm-yy
        /SINCE:dd-mmm-yy/THROUGH:dd-mmm-yy
        /TODAY
        /EXCLUDE:filespec
        /NOWARNINGS
        /SHARED

    The TYPE command displays the contents of text files on your
    terminal.

    $
```

The HELP text consists of a brief explanation of the command followed by an illustration of the syntax, showing that TYPE accepts one or more file specifications, and one or more qualifiers. See Section 4.4.1 for more information on TYPE and its qualifiers. Or, type a HELP command naming the qualifier to get more information on that qualifier.

```
    $ HELP TYPE TODAY

     TYPE/TODAY filespec[s]

     The /TODAY qualifier specifies that you wish the TYPE command to
     type only files created today.
```

If you want help while being prompted by the TYPE command, use the following procedure:

```
$ TYPE (RET)
File(s)? ? (RET)
 TYPE[/qualifier[s]] filespec[s]
        /DATE:dd-mmm-yy
        /SINCE:dd-mmm-yy
        /THROUGH:dd-mmm-yy
        /SINCE:dd-mmm-yy/THROUGH:dd-mmm-yy
        /TODAY
        /EXCLUDE:filespec
        /NOWARNINGS
        /SHARED

    The TYPE command displays the contents of text files on your
    terminal.

    File(s)? (RET)
```

The same help text is printed on your terminal, but the prompt returns, meaning the TYPE command is still waiting for you to list the files you want typed.

You can also get help on a specific subtopic while being prompted by a command by responding to the prompt with a question mark. For example, after getting help on SET, you can also get help on a specific function of SET:

```
    $ SET (RET)
    Function? ? (RET)
     SET thing
```

The SET command can be used to set something. The following things can be set with this command:

```
[DAY]TIME      DEBUG          DEFAULT     DEVICE
GROUPFLAGS     HOST           LIBRARY     [NO]PARTITION
PRIORITY       PROTECTION     QUEUE       SYSTEM
TERMINAL
```

For information on the SET ACCOUNTING command, see HELP ACCOUNTING SET.

To get help on a specific topic, enter a question mark followed by the thing:

    Function? ? DEFAULT (RET)

    SET DEFAULT [ddnn:][[ufd]]

    The SET DEFAULT command sets your default directory or device, or both.

    SET DEFAULT with no parameters returns a nonprivileged user to the login device and UFD. For privileged users, the unadorned SET DEFAULT returns to the login device, but the UFD remains the same.

    Function?   (RET)

You can also get help by typing a question mark in response to the dollar sign prompt ($) and to the explicit DCL prompt (DCL>).

If you should decide after reading the help text that you have chosen the wrong command, press CTRL/Z in response to the prompt to end the execution of the command. (A CTRL/Z in response to a prompt always cancels execution of a DCL command.)

From a DCL terminal, the command HELP provides information on DCL and the command HELP/MCR provides information on MCR. From an MCR terminal, the command HELP provides information on MCR and the command HELP/DCL provides information on DCL.

There may also be special help files providing information on special aspects of your installation. In addition, you can create local help files for your own use. See Chapter 3 for more information on the HELP command and help files.

1.2.4 MCR and DCL Commands

You can usually enter MCR commands from terminals set to DCL or DCL commands from terminals set to MCR.

If you are at a DCL terminal and want to enter an MCR command, simply precede the command with an MCR. The command is passed directly to MCR while the terminal remains set to DCL.

Here is the format of the MCR command:

    DCL>MCR
    MCR command line? mcrcommand

        or

    DCL>MCR mcrcommand

And, if you want to enter a DCL command from an MCR terminal, use the DCL command. The command is executed and the terminal remains set to MCR. Here is the format of the DCL command:

    MCR>DCL dclcommand

There are no prompts from this command.

1.2.5 Abbreviations

It is rarely necessary for you to type either the complete command name or the complete qualifier name. You only need to type the characters needed to distinguish the command or qualifier from all others.

For example:

- TYPE can be abbreviated T because it is the only command beginning with that character.

- INITIALIZE can be abbreviated INI, but not IN.

- INSTALL can be abbreviated INS, but not IN.

Three letters will usually be enough. Four letters will always be enough. You can often omit other parts of commands as well. You should experiment to find how short various commands can be cut. For instance, the following command:

    $ SET TERMINAL/VT100 (RET)

is the documented format for the command that sets a terminal as a VT100. The same command to VAX/VMS DCL does the same thing. However, the RSX-11M version of DCL permits you to type the following:

    $ SE VT100 (RET)

to achieve the same result. The second form does not work on VAX/VMS systems.

These briefer forms should be used interactively only, and not used when you are making a permanent record or creating an indirect command file. You should also see Section 1.7 for a discussion of compatibility between RSX-11M DCL and other versions of DCL used on other operating systems.

For your convenience, some frequently used commands have brief forms:

Command	Brief Form	Command	Brief Form
ABORT	A	HELP	H
BROADCAST	B	HELP	?
COPY	C	LOGOUT	LO
DIRECTORY	D	LINK	L
DEALLOCATE	DEAL	MACRO	M
DEASSIGN	DEAS	PRINT	P
EDIT	E	RUN	R
FORTRAN	F	SHOW	S

To save time and typing, use these brief forms to replace the command names when you are entering commands.

1.2.6 Numbers and Dates

DCL recognizes both octal and decimal numbers. You usually do not
have to identify a number as octal or decimal. All numbers are
recognized as being octal or decimal, unless otherwise noted.

DCL recognizes dates in two forms:

> dd-mmm-yy as in 21-JUN-85
>
> > or
>
> mm/dd/yy as in 6/21/85

System displays are always in the first format.

1.2.7 Multiple Parameters

Some commands permit you to enter a single file specification or other
parameter by a list of parameters. If you are entering a list of
parameters, each parameter must be set off by commas. For example:

> $ PRINT ALPHA.TXT (RET)

causes a single file to be printed, while the following command:

> $ PRINT ALPHA.TXT, BETA.TXT, GAMMA.TXT (RET)

causes three files to be printed. You have the option of including
blanks on either side of the comma in lists.

If you end the list with a comma, DCL prompts you for further
parameters. For instance:

> $ PRINT ALPHA.TXT, BETA.TXT, (RET)
> File(s)? GAMMA.TXT (RET)

Some commands accept a list of arguments to a single qualifier or
parameter.   In such cases, the list of arguments must be enclosed in
parentheses, with each element set off by commas.   For example, the
following command:

> $ MACRO/SHOW:(CALLS,COMMENTS) HIYA (RET)

If you need to enter only a single argument, you do not need the
parentheses. For example:

> $ MACRO/SHOW:CALLS HIYA (RET)

1.2.8 Underscore Character

The underscore character (_) is used to make DCL commands more
readable where two words are needed to name a single command element,
such as PRINT/FLAG_PAGE. However, you need not type the underscore to
enter the command. PRINT/FLAGPAGE is the same as PRINT/FLAG_PAGE.

1.2.9 Colon and Equal Sign

The command descriptions in this manual show arguments set off by a colon (:), such as follows:

    $ PRINT/COPIES:2 IZZY.TXT (RET)

You can always replace such colons with an equal sign (=), as in this example:

    $ PRINT/COPIES=2 IZZY.TXT (RET)

Colons in device names, such as DB1:, and so forth, cannot be replaced by equal signs.

1.2.10 Quoting Strings

If you want to include an exact string in a DCL command, put the string in quotes. For instance, the following message:

    $ BROADCAST/ALL Rock and roll will never die (RET)

is broadcast as follows:

    11-MAY-85 13:55   From PRINCE::WRITERS (TT64:) to ALL
    ROCK AND ROLL WILL NEVER DIE

while the following command:

    $ BROADCAST/ALL "Rock and roll will never die" (RET)

is broadcast as follows:

    11-MAY-85 13:56   From PRINCE::WRITERS (TT64:) to ALL
    "Rock and roll will never die"

You also need quotes when passing commands to tasks using the /PARAMETERS qualifier to MOUNT or the /COMMAND qualifier to RUN or INSTALL.

1.2.11 ANSI File Names

ANSI file names do not have the same format as Files-11 file names and may contain characters that are illegal in Files-11. For instance, "MONEY&." is a legal ANSI file name, as is "CASH&.". To use those file names in a command, simply enter them as shown in the following example:

    $ COPY (RET)
    From? MM:"MONEY&." (RET)
    To? MM1:"CASH&." (RET)

If you are copying a file from a tape to a Files-11 disk, however, you must use only Radix-50 file specifications, such as the following:

    $ COPY (RET)
    From? MM:"MONEY&." (RET)
    To? DB:[200,1]RADIX50.TXT (RET)

The Radix-50 character set consists of the numbers 0 through 9, the letters of the alphabet, the dollar sign ($) and the period (.).

1.2.12 DECnet and DCL

Some commands will accept DECnet node names as part of a file specification. They are as follows:

```
APPEND    CREATE      DIRECTORY    RENAME
COPY      DELETE      PRINT        TYPE
```

Several other DCL commands accept node names in the file specifications. However, DECnet modifies the basic syntax of these commands. These commands are: PRINT, SET PROTECTION, and SET FILE. Be sure to check the RSX DECnet Guide to User Utilities before attempting to use these commands.

If the node you select is part of your network, you can simply add the node to the file specification in the appropriate DCL command. Note that your terminal remains connected to your local node when you issue these commands to a remote node.

In the following example, you issue a command to type on your terminal the file ROMAN.TXT, which is located on remote node PRINCE and device DB2:, in directory [303,5].

```
$ TYPE PRINCE::DB2:[303,5]ROMAN.TXT RET
```

All the usual rules about file and volume protection are maintained, of course.

If the remote node you specify has a different style of file specification from RSX-11M, you must enclose the file specification in quotation marks, as shown:

```
$ TYPE EEETEE::"DISK$USERDISK:[DALTON]DESPERADO.DIS" RET
```

In addition to transmitting commands between nodes, DECnet also allows you to connect your terminal to a remote node. See the SET HOST command, described in the following section.

1.2.12.1 SET HOST

After you log in to a system, you can use the SET HOST command to connect your terminal to a different system.

Both your current system and the remote system must run DECnet software. In addition, you need to have an account on the remote system; otherwise, you will not be able to log in on the remote system after you issue the SET HOST command.

Format

```
SET HOST nodename
```

Parameters

nodename

> Specifies the name of the remote system that you want to connect your terminal to.
>
> A node is one system within a network of systems. The system that you originally log in on is called a local node; all other systems in the network are called remote nodes.

Once you have connected to the remote node, that operating system responds with a prompt. After you log in, use commands that the remote operating system accepts.

Type the LOGOUT command to log out of a remote node. After typing this command, you are relocated to your local node.

You can use SET HOST to connect to only one remote host at a time. For example, suppose you want information located at two different remote nodes, KING and JUNE. You cannot connect first to remote node KING, and then execute SET HOST again to connect to remote node JUNE. You must first log out of KING, which relocates you to your local node, and then use SET HOST to connect to JUNE.

See the RSX-11 DECnet documentation accompanying the DECnet software product for a full explanation of this command.

Examples

       $ SET HOST TOOTSI (RET)

       Connected to node "TOOTSI", System type = RSX-11M
       System ID: RSX TIMESHARING

       $ LOGIN (RET)
       Account or Name: 303,3 (RET)
       Password: (RET)
           .
           .
           .
       $

This SET HOST command connects your terminal to the remote node TOOTSI. The remote system identifies itself, then prompts you. You log in on the remote system by using the name and password of your account on that system.

1.2.12.2 SHOW HOST

The SHOW HOST command displays the name of the processor to which your terminal is connected currently. The display also shows you the name and version number of the operating system running on the processor.

The SHOW HOST command is most useful after you have connected your terminal to a remote system with the SET HOST command. However, SHOW HOST works whether or not your system runs DECnet software. Without DECnet on your system, this command simply displays information about your local operating system.

Format

       SHOW HOST

Example

       $ SHOW HOST (RET)

       HOST=TOOTSI RSX-11M V4.2 BL38

This example indicates the display from SHOW HOST. The name of your current processor is TOOTSI, which is running Version 4.2 of the RSX-11M operating system.

1.2.13 Command Line Continuation

Sometimes a command will not fit all on one line. In that case, you can continue the line by using the hyphen (-). When you end a command line with a hyphen and a carriage return, the DCL continuation prompt (->) indicates that you can continue entering the command line. If you are continuing a line from a prompt, such as **Task?** that prompt is the indication that the line is being continued.

This feature permits you to enter command lines including more characters than your terminal has room for on one line.

DCL commands are limited to 80 characters in all. When you type a continuation line, count the hyphen and the two times you press the RETURN key as three of the 80 characters. Of course, each blank, as well as each punctuation mark, counts as one character.

Here is an example of line continuation:

```
$ PRINT/COPIES:2/LENGTH:60/FLAGPAGE/AFTER:(04-JUL-85) OZY.TXT,- RET
->IZZY.TXT, FIZZY.TXT RET
```

The command is not entered until DCL encounters a line ending with a RETURN that is not preceded by a hyphen. In the example, the first carriage RETURN does not enter the command. Only the second RETURN enters the command. The RETURN can be on a line by itself.

1.2.14 Comments in Command Lines

You may want to include comments in command lines if you are keeping a permanent record, such as a command file, or simply an interactive session on a hardcopy terminal. You can include comments in a DCL command line with the exclamation point (!).

If the comment ends the command line, only a single exclamation point is needed, as in this example:

```
$ PRINT OZY.TXT !Poem by Shelley RET
```

If the comment is within the command line, two exclamation points are needed to set it off, as in this example:

```
$ PRINT!Parody of Shelley! IZZY.TXT RET
```

These comments are ignored and not interpreted in any way by DCL.

Comments can be placed at any natural break in the command line: between qualifiers, between parameters, even as part of a response to a prompt. Another example:

```
$ PRINT/FORMS:2!Letter-quality printer! IZZY.TXT RET
```

1.2.15 SET DEBUG

SET DEBUG displays the MCR translation of any DCL command.

Format

    SET [NO]DEBUG[/qualifier[s]]

Command Qualifiers

    /[NO]EXECUTE
    /FULL

Command Qualifiers

/[NO]EXECUTE

> Indicates that you wish the DCL command to execute after the
> translation is displayed. The default is /NOEXECUTE, meaning
> that the translation only is displayed. When SET DEBUG/NOEXECUTE
> is in effect, you can only issue SET DEBUG commands or MCR
> commands using the DCL MCR command; all other DCL commands are
> trapped and translated. You can issue SET NODEBUG to turn off
> SET DEBUG. You can also issue SET DEBUG with either /FULL or
> /EXECUTE or both.

/FULL

> Indicates that you wish SET DEBUG to display the logical symbols
> (IFORM queue) used by DCL to translate the command as well as the
> translation itself. See the RSX-11M/M-PLUS System Management
> Guide for information on how DCL uses these logical symbols.

Examples

        $ SET DEBUG (RET)
        $ SHOW TIME (RET)
        TIM
        $ INITIALIZE/EXTENSION:200/PROTECTION:(SYS:RWED,OWN:RWED,GRO,WO)(RET)
        Device? DK2 (RET)
        Label? HOTROD (RET)
        INI DK2:HOTROD/EXT=200./PRO=[RWED,RWED,,]
        $ SET DEBUG/EXECUTE (RET)
        $ SH TIM (RET)
        TIM
        21:33:46 11-FEB-85
        $ SET NODEBUG (RET)
        $

> In this example, the user first issued SET DEBUG. Then the user
> typed SHOW TIME. DCL displayed TIM, which is the MCR command to
> display the date and time, but no date and time is displayed
> because /NOEXECUTE was in effect. Then the user typed an
> INITIALIZE command and DCL displayed the MCR translation of the
> same command. Then the user typed SET DEBUG/EXECUTE and typed
> SHOW TIME again. This time, after displaying the MCR command
> TIM, DCL passed the command TIM to MCR and the date and time were
> displayed. Finally, the user typed SET NODEBUG to end session
> with SET DEBUG.

Notes

> Use SET DEBUG to check on the MCR syntax of any DCL command.
> This is particularly helpful when learning the brief forms of
> commands.

SET DEBUG creates your own copy of DCL that remains present while
SET DEBUG is in effect. Therefore, SET DEBUG creates a load on
the system. You should use SET DEBUG sparingly, particularly
when the system is under heavy use.

1.2.16 Errors

You can correct typing errors or change the line completely, by using
the RUBOUT or DELETE key, or by pressing CTRL/U (provided you have not
terminated the line).

You can get rid of any DCL command by pressing a CTRL/Z, provided you
have not terminated the command.

If the system detects an error in the command line input, it returns
the appropriate error message at the issuing terminal.

Here are some examples of incorrect commands and the error messages
they produce:

```
$ PRIJT IZZY.TXT (RET)
DCL  --  Illegal command

$ PRINT/PURPLE IZZY.TXT (RET)
PRINT  --  Illegal or contradictory qualifier
PRINT/PURPLE IZZY.TXT
      ^

$ PRINT/COPIES:TWELVE IZZY.TXT (RET)
PRINT  --  Numeral expected
PRINT/COPIES:TWELVE IZZY.TXT
            ^
```

In the first case, the error was detected by DCL, as indicated by the
first part of the error message. There is no DCL PRIJT command. The
entire command was rejected.

In the second case, the command was entered correctly, but the
qualifier was incorrect. The first part of the message shows that the
error was detected within the PRINT command itself. The command is
reprinted and a circumflex (^) points to the error.

In the third case, the command and qualifier were correct, but the
argument was in error. The message explains the error and the
circumflex points to the error.

Sometimes the circumflex does not point directly at the error, but at
the point at which the command started to go wrong, which may be
several characters before or after the actual error. Typing mistakes
are by far the most common cause of errors. Retyping the command
often eliminates the error. Other common causes of errors are
omitting a space or other delimiter in a command line, specifying
invalid devices or nonexistent files, issuing privileged commands from
a nonprivileged terminal, and failing to type a sufficient number of
characters to distinguish the command or command element.

The command descriptions include the most common errors produced by
the commands and suggestions for correcting the errors. All the DCL
error messages are listed and explained in Appendix A of this manual.

1.3 FUNCTIONAL GROUPING OF DCL COMMANDS

This manual includes seven functional groupings of commands, each grouping constituting a chapter. Each of these chapters is described here. Fully detailed descriptions of each command are included in these functional chapters. The functional groupings of commands are described in Sections 1.3.1 through 1.3.7. Each functional chapter also includes an introduction to the important concepts of the functional group.

Command descriptions can have five parts:

1. **Brief statement of the function** of the command.

2. **Full description of the format** of the command, including prompts, defaults, acceptable values for arguments, and the effects of each qualifier.

3. **Examples** of the command in use. These examples often illustrate less obvious aspects of the use of the command.

4. **Notes,** including warnings about side effects, counteracting commands, recommendations for further reading, and so forth. Some command descriptions do not have notes.

5. **Error messages,** including an explanation and a suggested user action for each error. Only error messages specific to the command are included in the command descriptions. General error messages, such as those relating to syntax, are described in Appendix A.

In addition, the formats of all RSX-11M DCL commands are presented in alphabetical listing in Chapter 9.

Nonprivileged commands are those commands needed in everyday use of the system by all users. Many privileged commands are also included in DCL. Privileged commands are those commands that affect system operations. For instance, the SHOW TIME command is a nonprivileged command, but the SET TIME command is privileged.

1.3.1 The SET and SHOW Commands

The SET commands dynamically alter system characteristics. The SHOW commands display system characteristics. These commands are introduced in Chapter 2. The most useful functions of SET and SHOW are also described in Chapter 2. However, because such a wide variety of functions can be altered or displayed, the remaining functions of SET and SHOW are listed in Chapter 2 but described in the appropriate functional chapter. SET and SHOW are the last commands described in each chapter. For example, SET DEVICES and SHOW DEVICES are described in full at the end of Chapter 5; and SET TERMINAL and SHOW TERMINAL are described at the end of Chapter 3.

The SET or SHOW command and its function are considered to form a single command. That is, rather than referring to the TERMINAL function of the SHOW command, you would refer to the SHOW TERMINAL command.

1.3.2 Terminal Operations

All commands directly related to terminal operations are described in Chapter 3. The commands include BROADCAST, HELP, LOGIN, LOGOUT, REQUEST, SET TERMINAL, and SHOW TERMINAL. In addition, Chapter 3 includes a detailed description of the use of the terminal keyboard on RSX-11M systems. All control (CTRL) characters and other important keys are explained.

1.3.3 Handling Files

Most information in RSX-11M systems is in files in User File Directories (UFDs). These files and directories are located within Files-11 volumes.

All commands relating to files are described in Chapter 4.

The commands include ANALYZE/MEDIA, APPEND, CONVERT, COPY, CREATE, CREATE/DIRECTORY, DELETE, DIFFERENCES, DIRECTORY, EDIT, EDIT/EDT, EDIT/KED, EDIT/SLP, PRINT, PURGE, RENAME, SORT, TYPE, and UNLOCK.

SET and SHOW commands described in Chapter 4 include SET and SHOW DEFAULT, SET FILE, and SET PROTECTION.

1.3.4 Handling Devices and Volumes

Volumes are held on magnetic media, usually disks. The volume is in the RSX-11M format, called Files-11. The disks are mounted on hardware devices. Devices can be as follows:

- Public - available to all users

- Shareable - available to some users but not others

- Private - available to a single user only

All commands relating to volumes and devices are described in Chapter 5. The commands include ALLOCATE, ASSIGN, ASSIGN/REDIRECT, ASSIGN/TASK, BACKUP, DEALLOCATE, DEASSIGN, DISMOUNT, INITIALIZE, INITIALIZE/UPDATE, and MOUNT.

SET and SHOW commands described in Chapter 5 include SHOW ASSIGNMENTS and SET and SHOW DEVICES.

1.3.5 Program Development

RSX-11M provides full facilities for program development, which are described in Chapter 6. The commands described include LIBRARY and LINK.

SET GROUPFLAGS and SHOW GROUPFLAGS are also described in Chapter 6.

1.3.6 Running Tasks

The task is the fundamental executable programming unit on RSX-11M. All commands related to running tasks are described in Chapter 7.

These commands include ABORT, CANCEL, CONTINUE, FIX, INSTALL, REMOVE, RUN, START, STOP/BLOCK, START/UNBLOCK, and UNFIX.

SET and SHOW commands described in Chapter 7 include SHOW COMMON, SHOW CLOCK_QUEUE, SET and SHOW PARTITION, SET PRIORITY, and SHOW TASKS.

1.3.7 System Control

System control commands are described in Chapter 8. Most of these commands are privileged, but the SHOW commands described in Chapter 8 are nonprivileged.

Only SET and SHOW commands are described in Chapter 8. These include SET and SHOW [DAY]TIME, SHOW MEMORY, SET and SHOW SYSTEM/DIRECTORY, SET SYSTEM/[NO]LOGINS, SET and SHOW SYSTEM/POOL, and SHOW USERS.

1.4 USING THE QUEUE MANAGER: PRINT JOBS

The RSX-11M Queue Manager provides facilities for printing files on line printers or other output devices. Files can be printed under user control or under the control of a system task or applications task. See the description of the PRINT command in Section 4.4.2.

All uses of the Queue Manager are described in a separate manual, the RSX-11M/M-PLUS Batch and Queue Operations Manual. The manual introduces and explains all the uses of the Queue Manager. The manual includes sample print jobs, all commands for placing jobs in queues and displaying queue contents, and a complete description of the process of setting up the Queue Manager. In addition, the manual includes an explanation of writing tasks to spool output.

1.5 INDIRECT COMMAND FILES IN DCL

In addition to the batch-processing subsystem, RSX-11M provides the Indirect Command Processor (Indirect) as an alternative means of automatically passing commands to the operating system. If you have a series of commands to be executed in the same or similar fashion every time, you can include these commands in a file to be run by Indirect.

Indirect accepts not only DCL and MCR commands, but also special directives that allow you to program the execution of the indirect command file. Indirect is described in full in the RSX-11M/M-PLUS Indirect Command Processor Manual.

An indirect command file can include both MCR and DCL commands. If the file contains commands to both of these command line interpreters (CLIs), you must either prefix each CLI command with the CLI name, or change the CLI setting of your terminal before each CLI command. When your file has completed, reset the terminal to your usual CLI in the indirect command file. For example, if your terminal is set to DCL and you have MCR commands in your indirect command file, you can prefix each MCR command with "MCR," or set the terminal to MCR before each command and then reset it to DCL afterwards.

1.6 DCL AND RSX-11M UTILITIES AND SYSTEM TASKS

The RSX-11M operating system includes a number of utilities and other system tasks designed to assist you in your use of the system. Most utilities are used to create or alter files in some way.

Many DCL commands invoke these utilities transparently to the user. Just as some DCL commands are passed to MCR, other DCL commands are passed to utilities.

One important utility is the Peripheral Interchange Program (PIP), which is used to manipulate files. DCL functions such as APPEND, COPY, DELETE, PURGE, and DIRECTORY depend on PIP. As another example, the LINK command invokes the RSX-11M Task Builder. Appendix B of this manual gives information on the relationships between DCL and the utilities and other system tasks.

For more information on each RSX-11M utility, see the RSX-11M/M-PLUS Utilities Manual. RMS-11 utilities are described in the RMS-11 documentation supplied with your system. Some system tasks that might be considered utilities, such as the Task Builder, are described in separate manuals.

For descriptions of other useful and important system tasks, see the following manuals:

- RSX-11M System Generation and Installation Guide - a description of the process of tailoring a system to a particular installation.

- RSX-11M/M-PLUS MCR Operations Manual - a description of MCR plus many details on system startup procedures and system operations.

- RSX-11M/M-PLUS Indirect Command Processor Manual - a full description of the Indirect Command Processor.

- RSX-11M/M-PLUS Batch and Queue Operations Manual - a description of the Queue Manager, which controls the orderly printing of files and other output, and the batch processing subsystem, a means of automatic use of the system. Batch processing is available on RSX-11M-PLUS systems only.

- EDT Editor Manual - describes the use of EDT, the DEC Standard editor. The manual includes an introduction to EDT.

- RSX-11M/M-PLUS Guide to Program Development - a detailed introduction to the program development facilities, both DCL and MCR, of RSX-11M/M-PLUS.

- PDP-11 MACRO-11 Language Reference Manual - an introduction to the MACRO-11 Assembly Language and Assembler.

- RSX-11M/M-PLUS and Micro/RSX Task Builder Manual - a description of the task building, or linking, process in RSX-11M.

1.7 RSX-11M DCL AND VAX/VMS DCL

DCL on RSX-11M Version 4.2 and subsequent releases is designed to be compatible with DIGITAL standard DCL on VAX/VMS Version 3.0 and subsequent releases. If the function being performed is identical, the same DCL command will have the same effect on an RSX-11M system or a VAX/VMS system. As other operating systems adopt DCL, the same will be true for them.

In most cases, the default forms of DCL commands will have the same effect on either kind of system. Furthermore, many qualifiers are the same and also have the same effect.

On the other hand, operating systems that implement DCL, including RSX-11M, will have system-specific extensions to DCL. For example, RSX-11M systems have several features not available on VMS systems; DCL commands have been implemented to invoke these features. The command syntax for these features is different from any VMS command syntax to avoid confusion.

In addition, many of the features discussed in this chapter are RSX-11M extensions to the basic DCL language. Prompts, HELP, error messages, system displays, and command parsing (including abbreviating) are unique to RSX-11M and may not be fully compatible with other DCL implementations.

NOTE

All the short forms and abbreviations in DCL are intended for the convenience of interactive users at terminals. You should not use either these brief command forms, or other abbreviations shorter than four characters per command element, in indirect command files or any other permanent form. As new commands are added to DCL, these abbreviations may change, but the full-format commands are stable.

CHAPTER 2

THE SET AND SHOW COMMANDS

The SET commands dynamically alter system characteristics.   The   SHOW
commands display system characteristics.

These commands affect every part of the   operating   system.   Some   of
them   are   needed   by   every user every day.   Others might not be used
once in a year.

SET and SHOW both prompt

    Function?

There are more than 20 functions that can be   altered   or   established
with   the   SET   command,   and more than 20 others, not all of them the
same, that can be displayed with the SHOW command. Most   of   the   SET
functions include options, values, or negations.

Customarily, the SET or SHOW command and the function   are   considered
to   form   a   single   command.   That   is,   rather than speaking of the
TERMINAL function of the SHOW command, you would   refer   to   the   SHOW
TERMINAL command.

Some of the most commonly used variations on the SET and SHOW commands
are described for your convenience in Sections 2.1 through 2.12.

Full descriptions of all the SET and   SHOW   command   functions   appear
later   in   this   manual.   Each chapter concludes with the SET and SHOW
commands appropriate to that chapter.   Thus, SET and SHOW TERMINAL are
described   at   the   end of Chapter 3, Terminal Operations, and SET and
SHOW DEVICES are described at the end of Chapter 5,   Handling   Devices
and Volumes.   See the Table of Contents for references.

The following are the   most   commonly   used   SET   and   SHOW   commands,
described   here   in   brief   form   for   your   convenience.   All of these
commands are described in detail elsewhere in this manual.

2.1 SET AND SHOW TIME

Format

    SET [DAY]TIME [hh:mm[:ss]]   dd-mmm-yy
                                  mm/dd/yy

You can set both the time and the date with this   privileged   command.
You   can   enter   the   date   in either form, for example, 04-JUL-85, or
7/4/85.   In the latter case, mm stands for month   number.   No   matter
how you enter the date, it will be displayed in the first form.

See Section 8.1.1.

The SHOW [DAY]TIME command displays the current setting of the day and time on your system.

Format

    SHOW [DAY]TIME

See Section 8.1.2.

2.2 SHOW MEMORY

    SHOW MEMORY

This command brings up the RMD display, which provides information on task activity, partitions, drivers, CPU usage, and so forth on your system.

Format

    SHOW MEMORY

On video terminals, this display is dynamic. On hardcopy terminals, the command prints a snapshot of the current RMD display.

The display provides a rough picture of the current state of the system. Alternate displays include the Active Task List or a single task header.

Not all systems include this command for interactive use. Most systems, however, do run the display on a slaved video terminal somewhere near the computer.

See Section 8.1.7 for more information on the memory display. See Section 7.12.4 for more information on the task list and task header displays.

2.3 SET AND SHOW TERMINAL

You can set and display many attributes of your terminal. The command

Format

    SHOW TERMINAL

without any qualifier, displays the current setting of all settable attributes of the terminal, as shown:

```
TT5:    [303,5]                 [303,5]
        CLI   = DCL     BUF  = 132.      HFILL = 0        SPEED=(300:300)
        LINES = 66.     TERM = LA34      OWNER = NONE     BRO    NOABAUD
        LOWER   NOPRIV  NOHOLD NOSLAVE   NOESC   NOCRT    HFF    NOREMOTE
        ECHO    NOVFILL NOHHT  NOFDX     WRAP    NORPA    NOEBC  TYPEAHEAD
        AVO     ANSI    DEC    NOEDIT    NOREGIS NOSOFT   NOBLKMOD
        NOHSYNC NOPASTHRU      TTSYNC    NOPRINTER_PORT
```

The display is explained in full in Section 3.8.2. Here, briefly, are the more significant fields:

Terminal number, login UIC, default UFD

CLI	Current CLI setting
BUF	Buffer-length, that is, the width in columns
SPEED	Transmit and receive speed
LINES	Page length in lines
TERM	Terminal model
OWNER	Tells whether terminal has been allocated
BRO	Terminal can receive broadcast messages
NOPRIV	Terminal is not privileged

The SET TERMINAL command permits you to set most of these attributes for your terminal.   Privileged users can set the attributes of any terminal.

Format

SET TERMINAL/attribute[/attribute[s]]

Note that you can set more than one attribute at a time.   A command such as the following:

$ SET TERMINAL/VT100/LOWERCASE/DCL (RET)

is accepted by the system and executed.

Here is a summary of the most important attributes you can set:

/DCL	/MCR	/UPPERCASE	/LOWERCASE
/PAGE_LENGTH:n	/WIDTH	/SPEED:(transmit,receive)	
/[NO]BROADCAST	/[NO]PRIVILEGE	/SCOPE	/HARD_COPY
/CLI=cliname	/HOLD_SCREEN	/INQUIRE	

You can also set your terminal as a particular model:

/ASR33	/LA36	/LQP02	/VT61
/ASR35	/LA38	/LQP03	/VT100
/DTC01	/LA50	/RT02C	/VT101
/KSR33	/LA100		/VT102
/LA12	/LA120	/VT05B	/VT105
/LA30P	/LA180S	/VT50	/VT125
/LA30S	/LA210	/VT52	/VT131
/LA34	/LN03	/VT55	/VT132
/VT200_SERIES	/PRO_SERIES		

See Section 3.8.1 for more information on the meaning of these attributes.

2.4 SHOW DEVICES

The SHOW DEVICES command displays the name and other information for all devices known to the system or, if specified, information for a specific device.

Format

SHOW DEVICES

This command displays the name and other information for all devices of a particular type known to the system, where dd is the 2-letter mnemonic for the device type or pseudo-device type. See Section 5.11.3.

2.5 SHOW QUEUE

This command displays information about the queue file maintained by the Queue Manager. Two forms of this command are most useful.

Format

        SHOW QUEUE

This command displays information about all jobs in all queues. In addition to showing all queue assignments, the information display includes all Queue Manager QMG jobs and all files included in those jobs.

        SHOW QUEUE/FULL

This command displays full information about all jobs in all queues. In addition to the information displayed by SHOW QUEUE, SHOW QUEUE/FULL displays all the attributes of all the QMG jobs in the queue, such as /FORMS, /COPIES, and /PAGE_LENGTH.

See the RSX-11M/M-PLUS Batch and Queue Operations Manual for more information on the SHOW QUEUE commands.

2.5.1 SHOW PROCESSOR

The SHOW PROCESSOR command displays information about the card readers, batch processors, printers, and other devices under the control of the Queue Manager.

Format

        SHOW processortype processorname[/qualifier[s]]

Parameters

processortype

        The processortype is one of the following:

                BATCH
                CARD_READER
                DEVICE
                INPUT
                PRINTER
                PROCESSOR

        CARD_READER and INPUT are synonyms. DEVICE refers to all nonbatch output processors and is a synonym for PRINTER. BATCH refers to all batch processors.

        A simple SHOW PROCESSOR command displays all processors in the system.

        See the RSX-11M/M-PLUS Batch and Queue Operations Manual for more information.

processorname

> For physical devices, the processor name is identical with the name of the physical device. Thus, the device LP1: is controlled by processor LP1:.

> Applications processors have names of six characters.

2.6 SHOW ASSIGNMENTS

The SHOW ASSIGNMENTS command displays at your terminal all local and login logical-device assignments. See Section 5.11.1.

Format

> SHOW ASSIGNMENTS[:ttnn:]

2.7 SET AND SHOW DEFAULTS

The SET DEFAULT command alters your default settings for device and directory.

Format

> SET DEFAULT [ddnn:] [g,m]

Parameters

> dd

>> The 2-letter device mnemonic

> nn

>> The 1- or 2-digit device number terminated by a colon

> [g,m]

>> The directory is two octal numbers separated by a comma and enclosed in brackets.

The default device name and directory are included automatically in every file specification included in a command unless overridden. See Section 4.5.1.

The SHOW DEFAULT command displays your current default device and directory. See Section 4.5.2.

Format

> SHOW DEFAULT

2.8 SHOW TASKS

This command displays information about tasks in the system. Four forms of the command are most useful.

The SHOW/TASKS/ACTIVE command displays information about the tasks active at your terminal.

Format

    SHOW TASKS/ACTIVE

The SHOW TASKS/ACTIVE/ALL command displays information about all tasks active on the system.

Format

    SHOW TASKS/ACTIVE/ALL

The SHOW TASKS/INSTALLED command displays full information about all tasks installed on the system.

Format

    SHOW TASKS/INSTALLED

The SHOW TASKS/INSTALLED/DEVICE command displays full information about all tasks installed on the specified device.

Format

    SHOW TASKS/INSTALLED/DEVICE:ddnn:

The following commands display full information on a single active or installed task.

Format

    SHOW TASKS/ACTIVE FULL taskname

    SHOW TASKS/INSTALLED FULL taskname

See Section 7.12.3.

2.9 SHOW USERS

The SHOW USERS command displays the terminal number and login UIC for each logged-in terminal.

Format

    SHOW USERS

See Section 8.1.6 for more information.

2.10 SET PROTECTION

The SET PROTECTION command alters the access protection of a file or files. If you wish to protect more than one file, the file specifications must be separated by commas.

Format

    SET PROTECTION filespec[s]
    Protection? (SYSTEM:RWED,OWNER:RWED,GROUP:RWED,WORLD:R)

Parameter

filespec

> The filespec parameter uniquely identifies a file. The file
> specification specifies the device name on which the volume
> containing the file is mounted, the directory in which the file
> is located, the file name, the file type, and the version number
> of the file.
>
> There are four kinds of access to files and four classes of users
> who may access files. The previous example shows the code for
> the default protection used on the system.
>
> The four classes of users who may access files are as follows:
>
> SYSTEM   Privileged users and the operating system itself.
>
> OWNER   Users logged in under the same User Identification Code
> (UIC) the file was created under.
>
> GROUP   Users logged in under the same group number as the UIC
> file was created under.
>
> WORLD   All other users
>
> The four kinds of access to files are as follows:
>
> R   READ access. The file can be typed, printed, copied,
> run (if a task image), and so on.
>
> W   WRITE access. The file can be written to or altered.
>
> E   EXTEND access. The file size can be changed.
>
> D   DELETE access. The file can be deleted.

The access you specify is the access that is given. If you specify
READ access to users in a particular category, those users will be
able to type, print, copy, or run (if it is a task image) your files.
However, your files are protected from being edited or deleted by
those users.

The protection status shown at the beginning of this section is the
default protection status accorded every file on the system. Use SET
PROTECTION to change the protection status of files.

You must enter the class of user and the type of access in the form
shown. If you do not wish to change the access rights of a particular
class of user, simply leave that class out.

All users can set the protection of their own files. Privileged users
can set the protection of any file. Note that you can set the
protection of a file so no user, including yourself, can do anything
with the file. If you do this, you will need a privileged user to
reset the protection when you wish to use the file.

You can display the protection of a file with the DIRECTORY command
and /FULL qualifier. See Section 4.3.1.

2.11 SET AND SHOW SYSTEM/DIRECTORY

The Executive, privileged system tasks, and other system files are kept together in a single directory called the system directory. In most cases, this will be the directory [3,54] on the system disk. An installation may, however, have more than one UFD containing such files, and, therefore, more than one system. For example, an installation may have one system for regular use and a second for backup or testing. The two systems can have quite different characteristics.

The SET SYSTEM/DIRECTORY privileged command sets the system directory as follows:

Format

        SET SYSTEM/DIRECTORY:[g,m]

The SHOW SYSTEM/DIRECTORY command displays the current system directory.

Format

        SHOW SYSTEM/DIRECTORY

On RSX-11M systems, the system directory also includes the utilities and other nonprivileged tasks.

The system directory on RSX-11M systems makes possible the RUN $taskname convention. When you issue a RUN or INSTALL command preceded by the dollar sign ($), you are instructing the system to look in the system UFD for a task image file of the name you have given, and to run or install the task from that directory. This convention saves you from having to find the files and specify the tasks by device and directory.

See Sections 8.1.3 through 8.1.5 for more information on these directories and their use.

2.12 SET DEBUG

The SET DEBUG command allows you to see the MCR translation of any DCL command.

Format

        SET [NO]DEBUG[/[NO]EXECUTE]

The default is /NOEXECUTE. This means that when you type a DCL command, the MCR translation appears on your terminal but the command is not executed. If you specify /EXECUTE, the translation appears on your terminal before the command is executed. See Section 1.2.15 for more information.

digital

PRINTED IN AUSTRALIA