DEC-11-SYDD-D

PDP-11

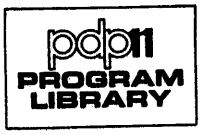
GETTING DOS ON THE AIR

OCTOBER 1971

THIS DOCUMENT IS FOR INFORMATION PURPOSES, AND IS SUBJECT TO CHANGE WITHOUT NOTICE.

IT DESCRIBES HOW TO LOAD THE DOS SOFTWARE ONTO DISK FROM PAPER TAPE AND DECTAPE. IT ALSO CONTAINS USEFUL PROGRAMMING INFORMATION.

YOUR ATTENTION IS INVITED TO THE SPECIAL NOTE ON THE NEXT PAGE.



S P E C I A L N O T E

SOFTWARE PERFORMANCE REPORT

If you have any problem or discover any inadequacy in your DOS software or its documentation, please report it using the Software Performance Report forms enclosed in your software kit.

Give the Software Performance Report to your DEC Software Specialist. In most cases he will be able to provide an immediate answer to your problem, as he is kept informed of new information as soon as it becomes available. If yours is an original problem, the Software Specialist will ensure that all necessary details, examples, and supporting material are included in the Report, and then he will forward the complete report to DEC's Software Information Service Group in Maynard, Massachusetts for a thorough investigation of your problem. As soon as the investigating programmer has the answer to your problem, it will be sent to you via the Software Specialist.

This procedure is intended to provide fast replies to your Software Performance Reports either by an immediate answer from your Software Specialist or as the result of concentrating our software maintenance effort on well-documented original problems.

Your inputs are most appreciated in our continuing effort to improve our software, and with your help our commitment to good software support will remain apparent.

If you have any questions on this procedure, please contact your Software Specialist.

READER'S COMMENTS CARD

Your attention is invited to the last page of this document. The "Reader's Comments" page, when filled in and mailed, is beneficial to both you and DEC; all comments received are acknowledged and are considered when documenting subsequent manuals.

PREFACE

Chapter 1 explains, using step-by-step instructions, how to build the PDP-11 Disk Operating System (DOS). We suggest that you read this chapter before building your first system. This reading should clarify the purpose for performing certain steps in the building process as well as familiarize you with the entire process.

Chapter 2 contains programming notes and techniques which reveal short-cuts and other helpful information for added programming efficiency.

Chapter 3 is a detailed description of the DOS Monitor DECtape Setup and Update Program (MODS). MODS can be used to copy the SYSLOD.SYS and MONLIB.SYS programs from the delivered DECtape onto a back-up DECtape.

This document assumes familiarity with the PDP-11 system, DOS Monitor, Link-11 Linker, and PIP File Utility Package.

DOCUMENTATION CONVENTIONS USED HEREIN

1. All command strings are terminated by typing the RETURN key; this is standard, and not shown in most examples since the key does not print on the teleprinter. Where necessary, the symbol ↵ is used to represent the RETURN key.

2. Monitor prints a period (.) or dollar sign($) to which the command string is typed. System programs print a number sign (#) to which the command string is typed.

   For clarity, these symbols have been underlined when the command strings they precede are typed by the user.

3. The CTRL/C key combination is typed by holding down the CTRL key while typing the C key; ↑C prints on the teleprinter. CTRL/C ensures that the Monitor will accept the next command, but will not necessarily immediately stop a job in progress.

4. In certain examples, sysdev: is used to represent the system device name of the particular installation's system disk, i.e., DK:, DF:, and DC:.

CONTENTS

CHAPTER 1

BUILDING THE DISK OPERATING SYSTEM

1.1 INTRODUCTION

The PDP-11 Disk Operating System (DOS) software is delivered on either paper tape or DECtape. The system software consists of the following programs, identified as shown below.

Monitor Modules	SYSLOD.SYS	
	MONLIB.SYS	
System Load Modules	PIP .LDA	(DECtape only)
	LINKOB.LDA	
	LINK11.LDA	
System Object Modules	PIP .OBJ	
	LINKOB.OBJ	
	LINK11.OBJ	
	EDIT11.OBJ	
	ODT11R.OBJ	
	LIBR11.OBJ	
	PALOB .OBJ	
	PAL11R.OBJ	
	PALSYM.OBJ	
	MODS .OBJ	(DECtape only)
Source Module	PALSYM.PAL	

(Building and running PDP-11 FORTRAN under the DOS is described in GETTING FORTRAN ON THE AIR, DEC-11-SFDB-D.)

The DOS Monitor is available in three major versions; each is tailored to run on a particular disk: RK11, RF11, or RC11. There are two versions for the RK11 disk: one for high-density (2200 BPI) and one for low-density (1100 BPI). The procedures for building the DOS differ slightly, depending on the disk that is to be used as the systems device. Section 1.4 describes how to load the system programs on the RK11 and RF11 disks. Section 1.5 describes how to load the system programs on the RC11 disk.

Briefly, the suggested sequence of operations in building DOS is:

1. Load the System Loader (SYSLOD.SYS).
2. Load and run the DOS Monitor (MONLIB.SYS).
3. Link the system object modules into load modules.

The newly-built DOS should be backed up by transferring a copy of the load and overlay modules onto paper tape or DECtape (see Section 1.6). MODS can be used to transfer the System Loader and Monitor from DECtape to DECtape (see Section 1.6.2 and Chapter 3).

This chapter contains operating procedures in a step-by-step fashion. It does not explain the purpose of each step as they are described in the DOS Monitor, Link-11, and PIP manuals.

1.2 LOADING THE SYSTEM LOADER AND DOS MONITOR

The System Loader (SYSLOD.SYS) is used to load and start the DOS Monitor (MONLIB.SYS). SYSLOD has been delivered on paper tape; and for systems with DECtape, it has been placed on DECtape for more automatic loading. To load the SYSLOD program into core, proceed to:

Section 1.2.1 for paper tape
Section 1.2.2 for DECtape

1.2.1 Loading SYSLOD from Paper Tape

The SYSLOD program is loaded from paper tape using the Bootstrap and Absolute Loaders:

1. Toggle the Bootstrap Loader into core using the console Switch Register as shown on the PDP-11 Instruction List card.
2. Set the ENABLE/HALT switch to HALT.
3. Place the Absolute Loader paper tape in the proper reader (as specified by the Bootstrap Loader). The special leader code (351) must be over the read sensors.
4. Set the Switch Register to xx7744, where xx is as specified in Step 1 above.
5. Depress the LOAD ADDRess switch.
6. Set the ENABLE/HALT switch to ENABLE.
7. Depress the START switch.

The Absolute Loader is read into core.

8. Place the SYSLOD.SYS tape in the proper reader with blank leader tape over the read sensors.

CAUTION

When placing paper tapes in the reader, ensure that blank leader tape is directly over the

the read sensors. When the name of the program is punched on the leader, blank tape separates the name from meaningful data. Exception:

Absolute Loader tape has special leader code 351 which must be over the read sensors.

9. Set the ENABLE/HALT switch to HALT.
10. Set the Switch Register to xx7500, where xx is as specified in Step 1 above.
11. Depress the LOAD ADDRess switch.
12. Set the ENABLE/HALT switch to ENABLE.
13. Depress the START switch.

The SYSLOD program is read into core and halts at location 30462. Any other halt is an error; start again at Step 8 above. (See Section 1.7 for an explanation of SYSLOD error halts.)

The SYSLOD program is in core. Go to Section 1.2.3.

1.2.2 Loading SYSLOD from DECtape

The SYSLOD program is automatically loaded from DECtape using the BM792-YB ROM Bootstrap Loader. (Users without the BM792-YB must load SYSLOD from paper tape.)

1. Mount the DECtape containing SYSLOD.SYS on DECtape 0.
2. Set the REMOTE and WRITE LOCK switches on DECtape 0.
3. Set the ENABLE/HALT switch to HALT.
4. Set the Switch Register to 173100.
5. Depress the LOAD ADDRess switch.
6. Set the ENABLE/HALT switch to ENABLE.
7. Set the Switch Register to 177344.
8. Depress the START switch.

The SYSLOD program is read into core and halts at location 30462. Any other halt is an error; start again at Step 1 above. (See Section 1.7 for an explanation of SYSLOD error halts.)

The SYSLOD program is in core. Go to Section 1.2.3.

1.2.3 Loading the DOS Monitor

There is one Monitor for the RF11 disk and one for the RC11 disk. However, there are two Monitors for the RK11 disk: a high-density

(2200 BPI) and a low-density (1100 BPI) version. Ensure that the Monitor is the one tailored for your disk.

CAUTION

Check to ensure that the disk write protection is disabled, i.e., not on. Operation of the write protection switch(s) is explained in the appropriate hardware maintenance manual. For RK systems, the cartridge must be on Unit 0.

Regardless of whether SYSLOD was loaded from paper tape or DECtape, the DOS Monitor (MONLIB.SYS) can be loaded from either paper tape or DECtape. Go to:

Section 1.2.4 for paper tape
Section 1.2.5 for DECtape

1.2.4 Loading MONLIB from Paper Tape

The DOS Monitor is on four paper tapes, labeled:

MONLIB.SYS V00 4A TAPE n OF 4

where n is the number giving the sequence in which the tapes are to be loaded. The loading procedure follows.

1. Set Switch Register bit 0 down (to 0 position).
2. Set Switch Register bit 15:
   down to replace the Monitor only
   up to clear (zero) the entire disk, which must be done when building on a new (fresh) disk.
3. Place the properly numbered MONLIB.SYS tape in the reader with blank tape directly over the read sensors.
4. Depress the CONTinue switch.

The tape is read in. With bit 15 up (to 1 position), the first MONLIB tape reads in about 4 inches and pauses while the disk is being cleared.

5. At end of tape, SYSLOD halts at location 30120. Any other halt is an error (see Section 1.7).
6. Repeat Steps 3, 4, and 5 for each MONLIB.SYS tape in the sequence specified by the tape number.

If additional paper tapes are to be loaded (Monitor patches, etc.), proceed to Step 7. If no additional tapes, go to Step 11.

7. Place the paper tape in the reader.
8. Depress the CONTinue switch.
9. At end of tape, SYSLOD halts at location 30120. Any other halt is an error (see Section 1.7).
10. Repeat Steps 7, 8, and 9 for each tape.

11. Set Switch Register bit Ø up (to 1 position).
12. Depress the CONTinue switch.

The Monitor is booted into core and prints:

```
MONITOR  V004A

$
```

Now log in as explained in Section 1.3.

1.2.5 Loading MONLIB from DECtape

When loading the DOS Monitor from DECtape:

1. Mount the DECtape containing MONLIB.SYS on DECtape Ø.
2. Set the REMOTE and WRITE LOCK switches on DECtape Ø.
3. Set Switch Register bit Ø up (to 1 position).
4. Set Switch Register bit 15:
   down to replace the Monitor only
   up to clear (zero) the entire disk, which must be done when building on a new (fresh) disk.
5. Depress the CONTinue switch.

With bit 15 up (to 1 position), the DECtape spins and then pauses while the disk is being cleared.

The System Loader loads the Monitor from DECtape to disk and halts at location 30120. Any other halt is an error (see Section 1.7).

If any paper tapes are to be loaded (Monitor patches, etc.), proceed to Step 6. If no tapes are to be loaded, go to Step 11.

6. Set Switch Register bit Ø down (to Ø position).
7. Place the paper tape in the reader.
8. Depress the CONTinue switch.
9. At end of tape, SYSLOD halts at location 30120. Any other halt is an error (see Section 1.7).

Repeat Steps 7, 8, and 9 for each paper tape.

10. Set Switch Register bit Ø up (to 1 position).
11. Depress the CONTinue switch again.

The Monitor is booted into core and prints:

```
MONITOR  V004A

$
```

Now log in as explained in Section 1.3.

1.3 LOGGING IN

Log in under User Identification Code 1,1 as shown below (The user types command strings in response to Monitor's . or $.):

```
$LOG 1,1
DATE:- 25-DEC-75
TIME:- 00:01:56
$
```

Monitor responds by printing a meaningless date and time. Use the DATE and TIME commands to enter the correct date and time. However, if the system does not contain a KW11-L Real-Time Clock, the TIME command is not necessary. For example:

```
$DATE 30-JUL-71
$TIME 16:42
$
```

You are now "on-the-air" with the DOS Monitor. System programs should now be loaded from paper tape or DECtape to the disk.

Section 1.4 for RK11 or RF11 disk
Section 1.5 for RC11 disk

1.4 BUILDING SYSTEM PROGRAMS ON THE RK11 OR RF11 DISK

The DOS system programs can be loaded on the disk from either paper tape or DECtape.

Section 1.4.1 for paper tape
Section 1.4.2 for DECtape

1.4.1 Building from Paper Tapes on the RK11 or RF11 Disk

Briefly, the process is as follows: Run the Link-11 Overlay Builder, LINKOB.LDA, and then the Link-11 load module, LINK11.LDA. Use Link-11 to link PIP.OBJ into a load module on the disk where it can be used to transfer the required system object modules from paper tape onto the disk. The object modules can then be linked into load modules using Link-11.

NOTES

Command strings should be typed in response to the underlined . and $ and # characters exactly as shown below unless otherwise indicated. All command strings are terminated with the RETURN key.

1. Place the LINKOB.LDA tape in the reader and type:

```
$RUN PR:
```

2. Place the LINK11.LDA tape in the reader and type:

   a. 8K systems:

```
$GET PR:
$SAVE LINK.LDA
$BE
```

   b. 12K to 28K systems:

```
$RUN PR:
```

Link-11 loads into core, starts automatically, and prints:

```
LINK-11 V005A

PASS 1
#
```

3. Place PIP.OBJ TAPE 1 OF 2 in the reader and type:

   a. 8K to 16K systems:

```
#PIP.LDA<PR:/CC/TA:2/E
```

   b. 20K to 28K systems:

```
#PIP.LDA<PR:/CC/TA:2/T:77474/E
```

CAUTION

Do not link PIP.OBJ higher than the top of 16K, i.e., with 20K or more, use the command string in b. above.

At end of tape, Link-11 prints:

```
A002 063320
$
```

4. Place PIP.OBJ TAPE 2 OF 2 in the reader and type:

```
$CO
```

Link-11 prints the load map (an example is shown below) and pauses:

```
TRANSFER ADDRESS: 022404
LOW LIMIT: 021612
HIGH LIMIT: 037460

PASS 2

A002 063320
$
```

5. Place PIP.OBJ TAPE 1 OF 2 in the reader and type:

```
$CO
```

Link-11 prints:

```
A002 063320
$
```

6. Place PIP.OBJ TAPE 2 OF 2 in the reader and type:

```
$CO
```

PIP.OBJ is now linked into a load module, PIP.LDA, on the disk. Link-11 prints:

```
LINK-11 V005A

PASS 1
#
```

7. Return to the Monitor by typing the CTRL/C key combination, clear core with the KILL command, and RUN PIP:

```
#^C
.KI

$RUN PIP


PIP-11 V004A

#
```

Use PIP to transfer the required object modules from paper tape to disk.

8. Place the paper tape (identified as shown in the command strings below) in the reader and type:

```
#LINKOB.OBJ<PR:/FB
#LINK11.OBJ<PR:/FB
#EDIT11.OBJ<PR:FB
#LIBR11.OBJ<PR:/FB
#PALOB.OBJ<PR:/FB
#PAL11R.OBJ<PR:/FB
#PALSYM.OBJ<PR:/FB
```

} (Not required for 8K systems)

9. Return to Monitor, clear core, and run Link-11:

```
#↑C
.KI
```

a. 8K systems, type:

```
$RUN LINK
```

b. 12K to 28K systems, place LINK11.LDA tape in reader and type:

```
$RUN PR:
```

Link-11 prints:

```
LINK-11 V005A

PASS 1
#
```

10. Link the required object modules to the top of available core (except PIP.OBJ) using the command strings shown below (use the TOP switch to link elsewhere).

NOTE

The load maps printed by Link-11 are not shown here since the format is similar for all links, and the data varies with the core size of the system and with the TOP switch when used.

```
#LINKOB<LINKOB.OBJ/E
#LINK<LINK11.OBJ/E
```

(Not required for 8K systems)

LINKOB and LINK11 must always be linked to the exact same HIGH LIMIT (see load maps).

```
#EDIT<EDIT11.OBJ/E
#LIBR<LIBR11.OBJ/E
#PALOB<PALOB.OBJ/E
#PAL<PALSYM.OBJ,PAL11R.OBJ/E
```

11. Return to Monitor, clear core, and RUN PIP:

```
#↑C
.KI
$RUN PIP
PIP-11 V004A
#
```

12. Direct PIP to list a brief directory of the files on the disk (an example is shown below):

```
#/BR

DF0:

MONLIB
LINK11.OVR
LINK  .LDA
PIP   .LDA
EDIT11.OBJ
LIBR11.OBJ
PALOB .OBJ
PAL11R.OBJ
PALSYM.OBJ
EDIT  .LDA
LIBR  .LDA
PALOB .LDA
PAL   .LDA
```

13. Delete all object modules from the disk:

```
#*.OBJ/DE
```

If you plan to debug user programs using ODT-11R, PIP the tape labeled ODT11R.OBJ from paper tape to disk:

14. Place ODT11R.OBJ tape in reader and type:

```
#ODT11R.OBJ<PR:/FB
```

15. Return to Monitor, clear core, and RUN LINKOB and PALOB:

```
#↑C
.KI

$RUN LINKOB          (Not required for 8K systems)

$RUN PALOB
```

16. RUN PIP and get a full directory of the newly-built Disk Operating System (an example is shown below):

```
$RUN PIP


PIP-11 V004A

#/DI


DIRECTORY DF0: [1,1]

00-XXX-72

MONLIB      508C 00-XXX-70 <377>
LINK11.OVR   36C 00-XXX-72 <233>
LINK  .LDA   60  00-XXX-72 <200>
PIP   .LDA   70  00-XXX-72 <233>
ODT11R.OBJ   44C 00-XXX-72 <233>
PAL11R.OVR   40C 00-XXX-72 <233>
EDIT  .LDA   56  00-XXX-72 <233>
LIBR  .LDA   35  00-XXX-72 <233>
PALOB .LDA  107  00-XXX-72 <233>
PAL   .LDA   65  00-XXX-72 <233>

TOTL BLKS: 1021
TOTL FILES:  10
```

Now run the newly-built DOS by assembling PALSYM.PAL (the source of the PAL-11R symbol table).

17. Return to Monitor, clear core, place the PALSYM.PAL tape in the reader, and assemble the source as shown below:

```
#↑C
.KI

$RUN PAL

PAL11R V005A

#PALSYM<PR:PALSYM.PAL


END

A002 063320
$
```

18. Place the PALSYM.PAL tape in reader again, type CTRL/C, and then the COntinue command:

```
$↑C
.CO

000000 ERRORS
```

The symbol table was assembled without errors. There is now a file named PALSYM.OBJ on the disk; delete it using PIP.

19. Return to Monitor, clear core, RUN PIP, and delete PALSYM.OBJ:

```
#↑C
.KI

$RUN PIP


PIP-11 V004A

#PALSYM.OBJ/DE
```

A back-up system should now be created as explained in Section 1.6.

1.4.2 Building from DECtape on the RK11 or RF11 Disk

Briefly, the process is as follows: Use PIP to transfer required object modules from DECtape to disk. Use Link-11 to link the object modules into load modules.

In the following examples an RF11 disk is assumed; with an RK11 disk, use DK0: wherever DF0: appears.

NOTES

Command strings should be typed in response to the underlined . and $ and # characters exactly as shown below unless otherwise indicated. All command strings are terminated with the RETURN key.

1. Mount the object module DECtape on Unit 0 (ensure that the REMOTE and WRITE LOCK switches are set on DT0:), then load and run PIP.LDA from DT0:

   a. 8K systems:

```
$GET DT0:PIP.LDA
$SAVE PIP.LDA
$BE
```

   b. 12K to 28K systems:

```
$RUN DT0:PIP.LDA
```

PIP prints:

```
PIP-11  V004A
#
```

2. Transfer required object modules from DT0: to disk:

   a. 12K to 28K systems:

```
#DF0:<DT0:PIP.OBJ,LINKOB.OBJ,LINK11.OBJ
```

   b. All systems:

```
#DF0:<DT0:EDIT11.OBJ,LIBR11.OBJ,PALOB.OBJ
#DF0:<DT0:PALSYM.OBJ,PAL11R.OBJ,MODS.OBJ
```

3. Return to Monitor by typing the CTRL/C key combination, clear core with the KILL command, and run LINKOB.LDA and LINK11.LDA:

```
#^C
.KI
```

a. 8K systems:

```
$RUN DT0:LINKOB.LDA

$GET DT0:LINK11.LDA
$SAVE LINK.LDA
$BE
```

b. 12K to 28K systems:

```
$RUN DT0:LINKOB.LDA

$RUN DT0:LINK11.LDA
```

Link-11 loads into core, starts automatically, and prints:

```
LINK-11  V005A

PASS 1
#
```

4. Link the required object modules to the top of available core (except PIP.OBJ) using the command strings shown below (use the TOP switch to link elsewhere).

NOTE

The load maps printed by Link-11 are not shown here since the format is similar for all links, and the data varies with the core size of the system and with the TOP switch when used.

CAUTION

Do not link PIP.OBJ higher than the top of 16K, i.e., with 20K or more, use the command string in b. below.

a. 12K and 16K systems:

```
#PIP<PIP.OBJ/CC/E
```

b. 20K to 28K systems:

```
#PIP<PIP.OBJ/CC/T:77474/E
```

c. All systems:

```
#LINKOB<LINKOB.OBJ/E
#LINK<LINK11.OBJ/E
```

(Not required for 8K systems)

LINKOB and LINK11 must always be linked to the exact same HIGH LIMIT (see load maps).

```
#EDIT<EDIT11.OBJ/E

#LIBR<LIBR11.OBJ/E

#PALOB<PALOB.OBJ/E

#PAL<PALSYM.OBJ,PAL11R.OBJ/E

#MODS<MODS.OBJ/E
```

5. Return to Monitor, clear core, run PIP, and delete all object modules from the disk:

```
#↑C
•KI

$RUN PIP

PIP-11  V004A
#*.OBJ/DE
```

6. Transfer ODT11R.OBJ and PALSYM.PAL from DT0: to disk:

```
#DF0:<DT0:ODT11R.OBJ,PALSYM.PAL
```

7. Return to Monitor, clear core, and RUN LINKOB and PALOB:

```
#↑C
•KI

$RUN LINKOB          (Not required for 8K systems)

$RUN PALOB
```

8. RUN PIP and get a full directory of the newly-built Disk Operating System (an example directory is shown below):

```
$RUN PIP

PIP-11  V004A

#/DI

DIRECTORY DF0: [1,1]

00-XXX-72

MONLIB      508C 00-XXX-70 <377>
ODT11R.OBJ   53  00-XXX-72 <233>
PALSYM.PAL   85  00-XXX-72 <233>
LINK11.OVR   36C 00-XXX-72 <233>
PAL11R.OVR   40C 00-XXX-72 <233>
PIP   .LDA   70  00-XXX-72 <233>
LINK0B.LDA   96  00-XXX-72 <233>
LINK  .LDA   59  00-XXX-72 <233>
EDIT  .LDA   56  00-XXX-72 <233>
LIBR  .LDA   35  00-XXX-72 <233>
PAL0B .LDA  107  00-XXX-72 <233>
PAL   .LDA   65  00-XXX-72 <233>
MODS  .LDA   30  00-XXX-72 <233>

TOTL BLKS: 1240
TOTL FILES:  13
```

Now run the newly-built DOS by assembling PALSYM.PAL (the source of the PAL-11R symbol table).

9. Return to Monitor, clear core, RUN PAL, and assemble PALSYM.PAL as shown below:

```
#^C
.KI

$RUN PAL

PAL11R  V005A

#PALSYM<PALSYM.PAL



 END

000000 ERRORS
```

The symbol table was assembled without errors. There is now a file named PALSYM.OBJ on the disk; delete it using PIP.

10. Return to Monitor, clear core, RUN PIP, and delete PALSYM.OBJ and PALSYM.PAL from the disk:

```
#↑C
.KI

$RUN PIP

PIP-11  V004A

#PALSYM.OBJ,PALSYM.PAL/DE
```

A back-up system should now be created as explained in Section 1.6.

1.5 BUILDING SYSTEM PROGRAMS ON THE RC11 DISK

Briefly, the Link-11 load module, LINK11.LDA, is run from DECtape and then used to link required system object modules from DECtape to disk. Note that DECtape is required when the RC11 is the only disk in the system.

NOTES

Command strings should be typed in response to the underlined . and $ and # characters exactly as shown below unless otherwise indicated. All command strings are terminated with the RETURN key.

1. Mount the object module DECtape on Unit 0 (ensure that the REMOTE and WRITE LOCK switches are set on DT0:), then load and run LINKOB.LDA and LINK11.LDA from DT0:

a. 8K systems:

```
$RUN DTØ:LINKOB.LDA

$GET DTØ:LINK11.LDA
$SAVE LINK.LDA
$BE
```

b. 12K to 28K systems:

```
$RUN DTØ:LINKOB.LDA

$RUN DTØ:LINK11.LDA
```

Link-11 loads into core, starts automatically, and prints:

```
LINK-11  V005A

PASS 1
#
```

2. Link the required object modules to the top of available core (except PIP.OBJ) using the command strings shown below (use the TOP switch to link elsewhere).

NOTE

The load maps printed by Link-11 are not shown here since the format is similar for all links, and the data varies with the core size of the system and with the TOP switch when used.

CAUTION

Do not link PIP.OBJ higher than the top of 16K, i.e., with 20K or more, use the command string in b. below.

a. 12K and 16K systems:

```
#PIP.LDA<DT0:PIP.OBJ/CC/E
```

b. 20K to 28K systems:

```
#PIP.LDA<DT0:PIP.OBJ/CC/T:77474/E
```

c. All systems:

```
#LINKOB<DT0:LINKOB.OBJ/E
#LINK<DT0:LINK11.OBJ/E
```

} (Not required for 8K systems)

LINKOB and LINK11 must always be linked to the exact same HIGH LIMIT (see load maps).

```
#EDIT<DT0:EDIT11.OBJ/E

#LIBR<DT0:LIBR11.OBJ/E

#PALOB<DT0:PALOB.OBJ/E

#PAL<DT0:PALSYM.OBJ,PAL11R.OBJ/E

#MODS<DT0:MODS.OBJ/E
```

3. Return to Monitor by typing the CTRL/C key combination, clear core with the KILL command, mount a fresh DECtape on DECtape Unit 2, and use PIP to ZEro DT2: (the back-up DECtape):

```
#^C
.KI
```

a. 8K systems:

```
$GET DT0:PIP.LDA
$SAVE PIP.LDA
$BE
```

b. 12K to 28K systems:

```
$RUN PIP
```

PIP loads into core, starts automatically, and prints:

```
PIP-11  V004A

#DT2:/ZE
```

4. Return to Monitor, clear core, run MODS, and copy SYSLOD.SYS and MONLIB.SYS from the supplied DECtape (DT0:) to the back-up DECtape (DT2:):

```
#↑C
.KI

$RUN MODS

MODS-11  V003A
#DT2:<DT0:
```

5. Return to Monitor, clear core, run PIP, and copy the following load modules onto DT2:

```
#↑C
.KI

$RUN PIP

PIP-11  V004A

#
```

a. 8K systems:

```
#DT2:<DT0:LINKOB.LDA,ODT11R.OBJ,PALSYM.PAL

#DT2:<PIP.LDA,LINK.LDA
```

b. 12K to 28K systems:

```
#DT2:<DT0:ODT11R.OBJ,PALSYM.PAL

#DT2:<LINKOB.LDA,PIP.LDA,LINK.LDA
```

c. All systems:

```
#DT2:<EDIT.LDA,LIBR.LDA,PALOB.LDA,PAL.LDA,MODS.LDA
```

6. Delete the following load modules from disk:

```
#LINK.LDA,EDIT.LDA,LIBR.LDA,PAL.LDA,MODS.LDA,PIP.LDA/DE
```

7. Return to Monitor, clear core, and run LINKOB.LDA and PALOB.LDA:

```
#↑C
.KI

$RUN LINKOB.LDA      (Not required for 8K systems)

$RUN PALOB.LDA
```

8. Run PIP from DT2:

```
$RUN DT2:PIP

PIP-11  V004A

#
```

a. 8K systems, delete PALOB.LDA from disk:

```
#PALOB.LDA/DE
```

b. 12K to 28K systems, delete LINKOB.LDA and PALOB.LDA from disk:

```
#LINKOB.LDA,PALOB.LDA/DE
```

9. List a full directory of the disk and back-up DECtape (example directories are shown below):

```
#DC0:,DT2:/DI

DIRECTORY DC0: [1,1]

30-JUL-71

MONLIB      284C 00-XXX-70 <377>
LINK11.OVR   36C 30-JUL-71 <233>
PAL11R.OVR   40C 30-JUL-71 <233>

TOTL BLKS:  360
TOTL FILES:   3
```

(directories continue on next page)

```
DIRECTORY DT2: [1,1]

30-JUL-71

SYSLOD.SYS     31C  30-JUL-71  <377>
MONLIB.SYS     84   30-JUL-71  <377>
LINKOB.LDA     24   30-JUL-71  <233>
ODT11R.OBJ     13   30-JUL-71  <233>
PALSYM.PAL     21   30-JUL-71  <233>
PIP    .LDA    17   30-JUL-71  <233>
LINK   .LDA    15   30-JUL-71  <233>
EDIT   .LDA    14   30-JUL-71  <233>
LIBR   .LDA     9   30-JUL-71  <233>
PALOB  .LDA    27   30-JUL-71  <233>
PAL    .LDA    17   30-JUL-71  <233>
MODS   .LDA     8   30-JUL-71  <233>

FREE BLKS:   282
FREE FILES:   44
```

The PDP-11 DOS is now ready for use. Run PAL from DT2: and assemble PALSYM.PAL (the source of the PAL-11R symbol table).

10. Return to Monitor, clear core, RUN PAL, and assemble as shown below:

```
#↑C
.KI

$RUN DT2:PAL

PAL11R  V005A

#PALSYM<DT2:PALSYM.PAL



 END

000000 ERRORS
```

The symbol table was assembled without errors. There is now a file named PALSYM.OBJ on the disk; delete it using PIP.

11. Return to Monitor, clear core, RUN PIP from DT2:, and delete PALSYM.OBJ:

```
#↑C
.KI

$RUN DT2:PIP

PIP-11  V004A

#PALSYM.OBJ/DE
```

1.6 CREATING A BACK-UP SYSTEM

A copy of the newly-built DOS can be transferred from disk to paper tape or DECtape. This back-up system is then available to facilitate building subsequent systems if necessary, or it can be used to transfer a fresh copy of any system load module into core or onto the disk.

1.6.1 On Paper Tapes

To be backed up with paper tapes, RUN PIP and transfer a copy of any disk file (excluding MONLIB and overlay files, OVR) onto punched paper tape. For examples:

```
$RUN PIP
PIP-11  V004A
#PP:<EDIT.LDA
#PP:<LIBR.LDA
#PP:<PAL.LDA
```

and etc.

Each paper tape punched should be labeled immediately after it is removed from the punch bin.

1.6.2 On DECtape

All files of the newly-built DOS can be transferred to a back-up DECtape. MODS is used to copy SYSLOD and MONLIB from DECtape to DECtape, and PIP is used to copy source, object, load, and overlay files.

1. Dial a DECtape transport to Unit 2.
2. Mount a fresh DECtape on DT2:.
3. Set the REMOTE and WRITE ENABLE switches on DT2:.
4. RUN PIP.
5. ZEro DT2: using PIP.
6. RUN MODS.
7. Copy SYSLOD.SYS and MONLIB.SYS onto DT2: using MODS.
8. RUN PIP.
9. Copy all or any disk file onto DT2: using PIP.

The printout might appear as shown below when transferring all disk files and SYSLOD and MONLIB onto DT2:

```
$RUN PIP

PIP-11  V004A

#DT2:/ZE

#↑C
.KI

$RUN MODS

MODS-11  V003A
#DT2:<DT0:
#↑C
.KI

$RUN PIP

PIP-11  V004A

#DT2:<*.LDA,*.OBJ,*.PAL,*.OVR/CO
```

Check the system device directory and copy only the existing file.extensions (*.PAL may not be required).

1.7 RECOVERING FROM SYSTEM HALTS

1.7.1 SYSLOD Error Halts

When a system loading error is detected, the SYSLOD program halts at the address shown in the console ADDRESS REGISTER with a display in the DATA register. The halts can be interpreted as follows:

Halt Address	DATA Display	
30222	2	Input data error.
	3	File open error (possibly a wrong tape was mounted).
	4	Load tape format error.
31570 (RK11 high) 31624 (RK11 low) 31620 (RF11) 31600 (RC11)		Read error occurred in trying to read the Monitor into core from disk.

Generally, when an error halt occurs the program must be reloaded. However, in the case of mounting the wrong tape (DATA=3), mount the correct tape (with MONLIB, start over with the first tape) and press the CONTinue switch twice--loading should resume correctly; otherwise, reload SYSLOD.SYS.

1.7.2 Halts While Linking

When the linking process is interrupted by a system error (Snnn), the cause is often due to:

1. The filename already exists on the directory. The user can use PIP to either specify a different filename in the command string or remove the filename from the directory with the UNlock and DElete commands, and then link again.
2. A bad read. The user can use PIP to UNlock and DElete the filename from the directory, and then link again.

Note that Link-11 enters the filename in the directory and then attempts to link the module.

1.7.3 Fatal Error Halts

When a fatal error message (Fnnn) is printed the system program is suspended. In most cases the BEgin command will restart the system program. If this fails, type CTRL/C followed by the KILL command and then restart the system program with the RUN command.

When the keyboard is locked, restart the Monitor as explained in Section 1.8.2.

1.8 RESTARTING THE MONITOR

1.8.1 From Keyboard

The user can, at any time, restart the Monitor with the FInish command. For example, when a programming session is concluded with the FI command, the Monitor is automatically restarted for another session. Unless the last character on the teleprinter is a $, the KILL command should precede the FI command. Printout might be:

```
#↑C
.KI

$FI
TIME:- 16:42:00

MONITOR  V004A

$
```

The user should then log in (see Section 1.3) under his UIC and issue the desired Monitor command. This type of restart does not alter file directories, date, or the time clock.

1.8.2 From Disk or DECtape

The BM792-YB ROM Bootstrap Loader can be used at any time to load a fresh copy of the Monitor into core from disk or DECtape. (Users without the BM792-YB must load SYSLOD from paper tape, as explained in Section 1.2.1.)

1. Set the Switch Register to 173100.
2. Set the ENABLE/HALT switch to HALT.
3. Depress the LOAD ADDRess switch.
4. Set the Switch Register to:

   177406 for RK11 systems
   177462 for RF11 systems
   177450 for RC11 systems

5. Set the ENABLE/HALT switch to ENABLE.
6. Depress the START switch.

A fresh copy of the Monitor is booted into core and prints:

```
MONITOR  V004A

$
```

Log in with the appropriate UIC. This type of restart does not alter file directories; date and time must, however, be re-entered.

A fresh copy of the Monitor can be booted into core from DECtape as explained from Section 1.2.2 onward.

CHAPTER 2

PROGRAMMING NOTES AND CAUTIONS

2.1. DOS MONITOR, VØØ4A

2.1.1 Programming Notes

1. The Monitor expects to be able to write on any part of a disk surface, therefore, the write protection must be disabled. In particular, the user should be careful to ensure that the appropriate switches are correctly set at times when there is no resident Monitor in core to produce the proper error diagnosis (i.e., during SYSLOD or a Monitor boot). Otherwise the system will not operate.

2. If it is desired to copy a DECtape containing SYSLOD, it is necessary to use either a block-by-block DECtape copy program or the MODS program (see Chapter 3). The nature of the SYSLOD file does not allow its being copied by PIP, i.e., it is a position-dependent contiguous file.

3. When the attention message AØØ2 is printed by the Monitor because of a DECtape not ready situation, it is safe to dial the DECtape (by rotating the thumb switch) to the proper drive number; unless the faulting request is OPENO, and the message is due to the requested drive being in WRITE LOCK status, and you wish to dial another transport to the drive number currently requested. In this one instance the directory of the DECtape has already been read by the Monitor and the Monitor is unaware that the dialing took place. Therefore, the directory of the latter tape will be garbled.

4. If CTRL/C is typed immediately following the RETURN key, the Monitor will echo the ↑C and hang -- a Monitor restart is required. A very slight hesitation (blink your eyes) after typing the RETURN key is all that is necessary to circumvent this problem.

5. It is legal to type a command to the Monitor at any time. However, if you are giving a command to the Monitor when it detects a situation requiring a diagnostic message (e.g., an OPEN failure in a running program while you are typing CTRL/C TI), a conflict for the swap buffer will result which causes the Monitor to hang. If this happens, a Monitor restart is required.

6. The use of the BEGIN command after a program failure requires some care. As far as possible the Monitor tries to clean up for the user by closing any input files currently open for input and deleting any files not yet completely created. The Monitor employs its own methods for doing this rather than using normal file-structured operations, because these operations would attempt to write to the relevant device data stored in core which must be suspect following the program crash. In particular, no attempt is made to change bit maps as permanently recorded. This can mean that on disks which use several bit maps to cover their surface, some blocks allocated to the files being deleted will not be released for further use, even though the files themselves have been removed. A series of crashes can thus lead to the disk being filled, although other evidence appears to contradict this. The user should therefore consider whether he should chance disk-corruption and use KILL rather than BEGIN and then delete the incomplete files with PIP. It also follows that if the automatic deletion effected by BEGIN could lose irrecoverable data to the user, he should again KILL after a crash.

7. If the system crashes to the extent that rebooting the Monitor from scratch is the only recourse, the user must be aware that when a file is opened for creation, its initial directory entry is made immediately. However, it is only when the file is closed that all the blocks allocated to it are permanently recorded as such. Because of the crash, the file may not be closed; some blocks given to it will still remain free and might be used for some other purpose. Nevertheless, the initial directory entry still exists and the only way to remove

it is by deletion, which includes release of all blocks associated with the file. If the deletion is not done immediately after the crash and before any other operations are carried out, it is possible that disk-corruption can ensue. It can be avoided only by the user doing the necessary clean-up as soon as convenient.

8. The Monitor includes drivers for RF11, RK11, and RC11 disks, and the CR11 card reader, as well as those available in earlier versions. The RF11 and RC11 Drivers are quite straightforward. The RK11 Driver expects all its associated units to be set up for high-density usage, except when the resident system device is the low-density cartridge. The CR11 Driver is suitable for reading only ASCII cards, though these may originate from either 026 or 029 punches; and the user has the option of requesting suppression of columns 73-80 and trailing blanks in preceding columns.

   However, the sources for both these drivers (which are available from DEC's Program Library) contain conditional assembly features to alter their usage if required. The Monitor also contains the necessary links for the driver for MT11 Magnetic Tape which will be available shortly.

9. Another edition of the DOS Monitor Programmer's Handbook (DEC-11-MWDB-D) is available. The user's attention is drawn particularly to the changes listed below:

   a. Change in SPECIAL FUNCTIONS CALL
   b. Extension of MODIFY command
   c. Introduction of new code under EMT 41 for requesting System Device Name
   d. Differentiation between ASCII and Binary in TRAN call
   e. Invalid code error in EMT 42

10. This release of the Monitor has the following discrepancies with the DOS Monitor Programmer's Handbook as published:

   a. No files are deleted on FInish, regardless of their protection codes

b. Macro calling for Monitor operations is not yet possible. The user must instead directly expand the sequences required in basic assembly language.
c. The only valid device accepted by console DUmp command is the line printer (LP:).

11. By its nature a dataset, other than one being updated, can supply input or accept output but not both simultaneously. Therefore, it is not possible to issue READ and WRITE commands to the same dataset at any one time and obtain meaningful results. Because a device, such as paper tape reader or punch, normally forces uni-directional requests or because the device is treated as file-structured, an appropriate OPEN command must be issued and this forces the direction. The Monitor provides the necessary checks in these cases. However, it does not presently protect the user of the Teletype, which is bidirectional and not file-structured. READ and WRITE on the same dataset will be accepted but will then cause invalid operations. The user must exercise caution by making two datasets available or, if the same dataset is used, logically changing the direction by : OPEN, READ, CLOSE, OPEN, WRITE, CLOSE.

12. The following bugs are currently known to exist in this version of DOS Monitor (VØØ4A).

a. When the system disk is RK11, an attempt to access a disk block outside the range of the disk surface (i.e., greater than 4800 on a high-density cartridge or 2400 on a low one) will cause the system to hang. If this should occur, reboot the Monitor.
b. If more than one output file is concurrently opened on an RK11 disk on any unit other than 0, it is highly likely that that disk's directory structure will be corrupted when the files are closed. For the present, users should restrict their usage of disks on these units to one output file at a time.
c. Occassionally it has been found that transferring card files to DECtape under PIP, where the blank supp suppress oftion for cards is exercised, can cause garbage data to be added to the end of the output file. Until this problem is rectified, the user can remove the garbage via the Editor.

The system is now loaded and operable. The user may issue keyboard commands and use the operating procedures described in Chapters 8 and 10 respectively. Users with new systems, that is, systems which have not previously been tailored, should refer to paragraph 7.4.2.

7.3.2.2 Bootstrap Restart Procedures - As mentioned

TABLE 2-1
Reserved Filename Extensions

Extension		Attribute
ALG	*	ALGOL source file
BAS		BASIC source file
BAK	**	Backup file
BLI	**	BLISS source file
CBL	*	COBOL source file
CIF	**	Core Image File
CMD		Command file
CRF		Input to cross-referencing program
DAT	**	DATA file for FORTRAN job
DDT		Reserved for DDT
DGN	**	Diagnostic message file
FTN	**	FORTRAN source file
FCL	*	FOCAL source list
LBO	**	Library of object modules (other types of libraries may also be implemented)
LDA	**	Load module, Absolute
LDR	**	Load module, Relocatable
LOG	**	Logging file
LSP	*	LISP source file
LST	**	Listing file
MAC		MACRO assembler source file
MAP	**	MAP file
MFD		Master file directory
OBJ	**	Ojbect module
OPR	**	Program generation information
OVR		Overlay
PAL	**	PAL assembler source file
PL1	*	PL/I source file
RNO	**	Input for RUN-OFF program
RPG	*	RPG source file
SNO	*	SNOBOL source file
SPC	**	SPEC format text
STB		Symbol Table (Link-11 output)
SYM		File of symbols
SYS		System management
TMP	**	Temporary scratch file
UFD		User file directory

Unstarred items may be used in the near future. No definite plans.

2.1.3 System Use of UIC's

The UIC is composed of two 8-bit fields and is represented as two 3-digit octal numbers whose values are less than or equal to 377. Thus, [2,316] is a legal UIC. The first field is referred to as the project number; the second field is the programmer number.

The following UIC's are reserved for use with DEC software. Installations should not use these UIC's for any purpose other than those purposes stated in DEC documentation.

Project numbers:	Ø-17	and	377
Programmer numbers:	Ø-17	and	377

Thus, the following are among those reserved:

[2,21Ø]	[1,1]	[Ø,Ø]
[17,377]	[4,17]	[Ø,15]

377 is reserved because * is mapped onto 377 by the Command String Interpreter.

2.1.4 Device Names in Radix-5Ø

The Radix-5Ø values for device names which appear on Monitor diagnostic message printouts are as follows:

Device	Radix-5Ø Value (Base 8)
DF	1476Ø
DK	1527Ø
DC	1457Ø
KB	4242Ø
PR	6332Ø
PP	632ØØ
LP	466ØØ
DT	16Ø4Ø
MT	5214Ø
CR	1262Ø
PT	6344Ø

2.3 EDIT-11 TEXT EDITOR VØØ2A

2.3.1 Programming Notes

1. Use of PT: and KB: -- When using KB: for input, commands to Edit-11 which read text into the buffer require that the user manually supply the end-of-page or end-of-data flags. The procedure is as follows: (↵) denotes the RETURN key):

   a. To end a Read command from KB: the user should type CTRL/C followed by

      END

      followed by the RETURN and LINE FEED keys in that order.

   b. To end an EXit command from KB: the user should type CTRL/C followed by

      END

      followed by the RETURN and LINE FEED keys in that order.

   c. When using PT: for input, the procedure is:

      (1) Put paper tape in reader and move switch to START.

      (2) Start Edit-11 and type command string.

      (3) Edit-11 will open the primary output file, read a few characters, then print an * to signal its readiness for a command.

      (4) Leave the reader on. Edit-11 will read tape when commanded to do so.

      (5) When the end-of-tape is reached during a read operation, signal so by typing CTRL/C followed by

      END PT

      followed by the RETURN and LINE FEED keys in that order.

      (3) When Edit-11 finishes punching leader, it pauses and waits for the punch to be turned off. When you have done so, pressing the LINE FEED key will resume Edit-11 operation.

      (4) From this point on, Edit-11 will pause before and after every output command to allow you to turn the punch on and

off. Pressing the LINE FEED key will cause it to continue.

d. When using PT: for output, the procedure is:

(1) With punch off, start Edit-11 and type command string.

(2) When Edit-11 opens the punch for output, it will begin outputting nulls to the teleprinter. Turn on the punch.

2. Monitor A Errors -- If at any time during the editing process, or during the initialization, an AØØ2 error is received from the Monitor, the user is required to take action to ready a device. Common causes are specifying PR: when there is no tape in the reader or specifying the wrong DECtape drive. Correct the problem, then type:

CO

followed by the RETURN key. Edit-11 will continue.

3. PT: Versus DF: -- If the user plans to do a lot of paper tape editing, he can often save a lot of time by using the disk to store edited versions of files. Edit from PT: to DF: then do editing from disk to disk. When satisfied, PIP the good file back to paper tape.

4. Use of Subsidiary Output -- As mentioned in the Edit-11 manual, use of a fast output device such as LP: for listing text via the EW command will greatly increase editing turn-around time. Use the L command sparingly and the printer often.

2.3.2 Cautions

1. If Edit-11 encounters an invalid Open because of a file already existing, it does not recover. Rather, an FØ12 Monitor error will be printed on the teleprinter, and it is necessary to restart the Editor with CTRL/C and BEgin.

2. The F and T commands presently provide for outputting only in the primary output device. To provide a Form Feed on the secondary output file, the Form Feed character must be inserted into the page buffer by the user. There is also presently no means of punching trailer on the secondary output device.

2.4 ODT-11R DEBUGGING PROGRAM VØØ2A

2.4.1 Caution

A problem exists in version VØØ2A and previous versions with regard to the saving of breakpoints at ODT's three entry points (OD K; OD R; OD). There are two substantially different ways that ODT can be entered:

1. ODT was used to set breakpoints in a program and the program crashed when it was given control by ODT. In this case, the Monitor will have control and the user will have typed:

   OD K

   or the user will have manually restarted ODT.*

2. Any other entry to ODT (via Monitor commands such as RV, BE, OD; or because a breakpoint occurred).

VØØ2A does not distinguish between these cases. Consequently, when ODT saves breakpoints and restores user instructions to their previous values, it will restore memory correctly in case 1 , but will move an octal ØØØØ3 into locations on which breakpoints have been set in case 2. This will be fixed in a subsequent release of ODT.

Until then, avoid exiting from ODT to the Monitor with breakpoints set in the program if ODT is to be re-entered. For example: when patching and saving prior to debugging:

```
$GE  prog
$OD
ODT11R  VØØ2A
*                          (make patches here)
  .
  .
  .
$SA
$OD K                      (reenter ODT, preserve re-
                           location registers)
BEnnnnnn
*                          (Set breakpoints here)
  .                        (Debug - if crash occurs,
  .                        exit OD K)
  .                        (make additional patches)
  .
  .
$;B                        (clear breakpoints)
*C
$SA                        (save includes ODT)
$OD K
BEnnnnnn                   (reenter ODT)
  .                        (continue using ODT)
  .
  .
```

*ODT's three starting addresses are at ODT's base $+172_8$(OD), 174_8(OD R), and 176_8(OD K).

CHAPTER 3

MONITOR DECTAPE SETUP AND UPDATE PROGRAM

3.1 INTRODUCTION

The Monitor within the PDP-11 Disk Operating System (DOS) is basically a series of individual program modules, each of which can satisfy some requirement of a user program or of an operator request from the console keyboard. The appropriate module is normally brought into the computer memory only when needed. Otherwise, it resides within a library (MONLIB) on the system disk and for speed of loading it must be stored as a core image.

It is the function of the System Loader (SYSLOD), generally, to prepare the system disk for DOS usage and, in particular, to establish MONLIB in the required form. It accomplishes the latter by converting the Monitor modules from the load format produced by PAL-11R Assembler and Link-11 Linker processes. If the modules are linked to paper tape, they can be fed directly to SYSLOD in a specified order via the PC11 Reader. SYSLOD, however, is also prepared to accept as input a DECtape file, MONLIB.SYS, which contains the load-format modules already strung together in the required sequence.

The Monitor DECtape Setup and Update Program (MODS) provides a means whereby the user can prepare this DECtape file and later maintain it, should existing modules need to be replaced or new ones be added. Moreover, the ROM Bootstrap, stipulated as a basic element of any DOS-supplied configuration to enable automatic Monitor startup from disk, can also be used with DECtape. MODS, therefore, stores SYSLOD itself on the same tape as MONLIB.SYS as a core image readily accessible by the Bootstrap. The basic format of the tape thus produced uses facilities outside those provided by the standard file structure operations of DOS. It cannot, as a result, be copied by the usual means, PIP-11 (although once prepared it can thereafter be used to store other files in the normal way). Instead, MODS can be used to copy the basic system tape content to another tape.

3.1.1 Operating Configuration

The MODS program is designed to operate under DOS. It can be run on any configuration which will support DOS, provided that DECtape is included. For ease of operation, however, a PC11 Reader is also

recommended. The ASR-33 Teletype as the sole means of paper tape input is time-consuming and imposes other restrictions which will be discussed later.

3.1.2 Outline of This Chapter

This Chapter is intended to show how MODS may be used. Section 3.2 gives a general description of the program and Section 3.3 details its operating instructions.

It is assumed that the user is already familiar with the general principles of DOS as described in the Programmer's Handbook (DEC-11-MWDA-D).

3.2 GENERAL DESCRIPTION

3.2.1 Command Input

A command string entered through the console keyboard directs the MODS program to operate in one of two modes and at the same time indicates the relevant devices to be used.

a. Setup -- A new DECtape is prepared to contain SYSLOD and MONLIB.SYS by transferring both from the same source, either DECtape or paper tape(s); or by copying MONLIB.SYS from another DECtape and inserting a new SYSLOD from paper tape.

b. Update -- The new DECtape is a copy of SYSLOD from another DECtape and MONLIB.SYS is also a copy but, in this case, possibly modified by new module versions entered from paper tape.

The command strings to accomplish this are shown in detail in section 3.3. In general, since MODS uses the DOS Monitor Command String Interpreter (CSI) to analyze the user's input, they follow the format prescribed for that routine.

MODS checks the validity of the user's specification and, if any error is encountered, an appropriate message is printed at the console teleprinter and the user is requested to enter a new string. MODS also confirms, where SYSLOD is to be copied from another DECtape, that it is, in fact, available on that tape. If not, a Monitor fatal error message results and the user must begin MODS anew.

3.2.2 Basic Preparation

The first step in the preparation of the new DECtape for either mode of operation is to establish a basic file-structure upon it. Basically this requires the writing of appropriate information in Blocks 100-101 (Master File Directory), Blocks 102-103 (User File Directory), and Block 104 (Master Bit Map). In addition, Blocks 70-77 are cleared in readiness for later usage as storage for individual File Bit Maps.

While providing this basic layout, MODS also reserves two contiguous areas on the new tape and enters appropriate identifying file information in the UFD:

a. Blocks 0-37 will be used to store the core image of SYSLOD (covering memory locations 0-36776) and its Loader called by the ROM Bootstrap.

b. Blocks 320-1077 are set up as a dummy file to force the allocation of blocks to MONLIB.SYS to be at the forward end of the tape only.

At the same time, MODS initiates the filling of its internal buffers with any data to be entered from paper tape.

3.2.3 Transfer of SYSLOD

In either mode, SYSLOD can be copied from another DECtape by merely transferring Blocks 0-37 from that tape to the new tape. This is speedily done by moving alternate sets of four blocks in two tape travel directions.

The paper tape SYSLOD is in the load format output by the Link-11 Linker. It must be preceded by a special resident version of DOS Monitor in similar load format. (The DEC Program Library supplies both routines on the same tape.) This format must be appropriately converted into the core image required. Moreover, in order that this image can be accessible via the ROM Bootstrap, an appropriate loader must be provided.

Therefore, MODS first moves into Block 0 the required loader from a copy stored within itself. It then checks that the paper tape being entered does, indeed, contain the special Monitor version and, if so converts the input into its core image starting at Block 1 on the tape

and continuing across contiguous blocks, two at a time. When the end of this routine is encountered, MODS expects the entry of SYSLOD itself and checks for this. If satisfied, the similar core image on the tape is prepared with blocks corresponding to any unused intervening core space first being cleared. MODS also establishes an appropriate link between the two routines within the image.

During this entire transfer process, errors arising either because the tapes entered are not those expected or are not in the correct format or through general read failure are reported to the user by an appropriate message at the printer and a new start must be made.

3.2.4 MONLIB.SYS Entry (Setup Mode)

When stored on the DECtape, MONLIB.SYS is a normal linked file within the DOS file-structure scheme. To copy from one DECtape to another is a relatively straightforward operation, provided that the input tape specified by the user does, in fact, contain the requisite copy. Because of the dummy file mentioned in section 3.2.2, the second half of the transfer between the two tapes will occur with the new tape moving in the reverse direction, finishing approximately back at the Directory blocks.

When MONLIB.SYS is merely a string of the Monitor modules in linker format, copying from paper tape is a simple process. In this case, however, the user must ensure that the first module entered is a copy of the normally permanent resident Monitor and the second module is the transient Monitor section TMON linked with a copy of the READ-WRITE processor. The remaining modules may follow in any order. (Tapes supplied by DEC Program Library are numbered to provide the requisite sequence.)

3.2.5 MONLIB.SYS Entry (Update Mode)

The user may be completely unaware of the sequence in which MONLIB.SYS was originally set up on this input DECtape. To allow him to enter his new modules from paper tape in any order, the Update mode of MODS first stores all of them as a series of temporary files on the system disk and maintains its own in-core directory of these for easy access later.

The following points concerning this process should be noted:

a. MODS uses the loader format to differentiate between modules. Therefore, there is no need for the user to enter them individually; if required, they can all be strung together on the same physical paper tape.

b. Should an error occur at this stage, some of these disk files could remain and might then cause file-structure errors if MODS is asked to start over. To prevent this, MODS first deletes any file on the disk using a name it is about to enter. Users should, therefore, avoid setting up their own files using names allocated to Monitor routines as given in the programming manual with a .TMP extension.

c. It follows from b. that if the user enters more than one copy of the same module, the last one seen will always be the one included in the final version of MONLIB.SYS on the new DECtape.

After all the paper tape modules have been handled in this fashion, MODS begins its transfer to the new DECtape. As the start of each module is encountered on the input tape, the in-core directory is searched for a corresponding new version. If none has been supplied, the original is merely copied across. Otherwise, the original is skipped and the new version is used to replace it. The disk file is then immediately deleted, but its in-core reference is retained. Because the skip of the original precedes its replacement from disk and MODS ignores the latter if the required file is no longer present, the immediate deletion has the effect of removing any other copy of the same module which might occur later within the original DECtape file. Thus the new MONLIB.SYS file may be not only an updated version but also a tidier one.

Finally, at the end of the original DECtape input, MODS makes a last search of its in-core directory and adds any completely new modules to the output file.

3.2.6 General Cleanup

After closing the new file, MODS requests deletion of the dummy file at the end of the tape and reinitializes itself for a further run. The new tape itself can be used later to store other system programs using PIP.

3.2.7 General Comment on Paper Tape Usage

The MODS program has been designed to allow the use of either PC11 Reader, if available, or the paper tape reader of ASR-33 Teletype, in that transfers are carried out using .TRAN requests to the Monitor rather than the more customary .READ. This follows from the fact that the ASR-33 is classified as an ASCII-only device and cannot be used to read the binary data of a Linker output module.

However, another problem still exists. All input from the ASR-33 is monitored for the start of a keyboard command from the operator, as indicated by the entry of CTRL/C. Because the computer cannot distinguish between a character coming from the keyboard and one coming from the paper tape reader, recognition of the appropriate ASCII code (3 or 203) within the binary data is treated in this way. As a result, MODS cannot allow operator intervention if its paper tape input comes from the ASR-33.

3.3 OPERATING INSTRUCTIONS

3.3.1 Loading MODS

The new DECtape to be prepared by MODS must be set up under the standard System Identification Code 1,1. The user should, therefore, ensure that he has logged in under this. He may then request MODS loading by a normal RUN command at the keyboard. When entered, MODS identifies itself and prints # in readiness for the input of its command string. Assuming MODS has been stored on the disk, the resulting teleprinter record might be (user input underlined):

```
MONITOR   V002A
$  LO 1,1
DA: dd-mmm-yy
TI: hh:mm:ss
$RU MODS

MODS-11   V003A
#
```

3.3.2 Command Input

As indicated in the General Description, section 3.2.1, the input string now entered by the user can be one of several forms, depending upon requirements. Examples are given below:

NOTE

1. [...] Items enclosed in square brackets can be omitted from the string, in which case DTØ: will be the default assumption.
2. {...} Items enclosed in braces show possible alternatives, one of which must be present.
3. n represents the DECtape unit number.
4. Ta:n indicates the decimal number (n) of physical paper tapes to be entered via PC11 Reader and must be supplied.

a. Set up a new DECtape where MONLIB.SYS and SYSLOD are on the same source:

$$[DTn:<]\begin{Bmatrix} DTn: \\ PR:/TA:n \\ PT: \end{Bmatrix}$$

b. Set up a new DECtape with MONLIB.SYS from another DECtape and SYSLOD on paper tape:

$$[DTn:<]\ DTn:,\begin{Bmatrix} PR:/TA:n \\ PT: \end{Bmatrix}$$

c. Update MONLIB.SYS from another DECtape, new modules being entered from paper tape:

$$[DTn:]/U<DTn:,\begin{Bmatrix} PR:/TA:n \\ PT: \end{Bmatrix}$$

Upon completion of the requested operation, MODS prints #.

3.3.3 Error Reporting

Errors in the input string or through read failure in the subsequent transfer operations are diagnosed by plain language messages. In all cases, MODS returns to request a completely new input string by printing # following the message.

File structure errors due to the nonexistence of SYSLOD or

MONLIB.SYS on a specified input DECtape result in a fatal diagnostic message from the DOS Monitor. The user may then request a restart by either a Begin or a REstart command.

3.3.4 Operator Action During Processing

Apart from the special treatment required if PT: forms part of the input specified (see section 3.3.5), the only operator actions likely to arise during processing are as follows:

a. A specified DECtape is not ready -- the DECtape driver initiates the printing of an A002 16040 message if the correct units are not made available or the output unit is write-locked. The program is resumed if the omission is rectified and CO) is entered as a keyboard command.

b. At the end of each physical paper tape being entered via the PC11 Reader (until the count given in the input string runs out), MODS prints an A002 63320 message. The next tape should be placed in the reader and CO) be entered.

NOTE

MODS expects the first paper tape to be already set up in the reader when the command input string is issued. If not, the nonexistent tape at this point counts as one of the entries.

3.3.5 Special Action for PT: Usage

As noted in the General Description under Section 3.2.7, usage of PT: poses a special problem. For this version of MODS (V003A) this can only be solved by the user making the following modifications to the resident Monitor before entering the input command string:

```
#↑C                     ;SUSPEND PROGRAM & ENTER MONITOR
.WA                     ;          LISTENING MODE

$ MO 3374               ;CHANGE MONITOR AS REQUIRED

3374/1011      411      ;USER TYPED 411

$ MO 3424
3424/1767      412      ;USER TYPED 412

$CO                     ;RESTART MODS OPERATION
                        ;NOW ENTER COMMAND STRING
```

After typing in the command string as required, the paper tapes can be fed through the ASR-33 reader one by one (with the reader turned on). No computer pauses occur between tapes. Also, the user should avoid striking the keyboard during this period, as all input is now deemed to originate from the reader. It is also advisable that the user take extra precautions to avoid error action necessitated by failure to set up the DECtapes correctly.

When all paper tapes have been input, the user should halt the computer and manually restore the locations changed above. He should then press CONTinue and enter the following command at the keyboard in order to proceed:

```
#↑C
.EN PT:
```

followed by the HERE IS key on ASR33 or BREAK key on ASR35.

NOTE

Tapes should be removed from the reader while blank trailer tape is still being read.

HOW TO OBTAIN SOFTWARE INFORMATION

Announcements for new and revised software, as well as programming notes, software problems, and documentation corrections are published by Software Information Service in the following newsletters.

Digital Software News for the PDP-8 & PDP-12
Digital Software News for the PDP-11
Digital Software News for the PDP-9/15 Family

These newsletters contain information applicable to software available from Digital's Program Library, Articles in Digital Software News update the cumulative Software Performance Summary which is contained in each basic kit of system software for new computers. To assure that the monthly Digital Software News is sent to the appropriate software contact at your installation, please check with the Software Specialist or Sales Engineer at your nearest Digital office.

Questions or problems concerning Digital's Software should be reported to the Software Specialist. In cases where no Software Specialist is available, please send a Software Performance Report form with details of the problem to:

Software Information Service
Digital Equipment Corporation
146 Main Street, Bldg. 3-5
Maynard, Massachusetts 01754

These forms which are provided in the software kit should be fully filled out and accompanied by teletype output as well as listings or tapes of the user program to facilitate a complete investigation. An answer will be sent to the individual and appropriate topics of general interest will be printed in the newsletter.

Orders for new and revised software and manuals, additional Software Performance Report forms, and software price lists should be directed to the nearest Digital Field office or representative. U.S.A. customers may order directly from the Program Library in Maynard. When ordering, include the code number and a brief description of the software requested.

Digital Equipment Computer Users Society (DECUS) maintains a user library and publishes a catalog of programs as well as the DECUSCOPE magazine for its members and non-members who request it. For further information please write to:

DECUS
Digital Equipment Corporation
146 Main Street, Bldg. 3-5
Maynard, Massachusetts 01754

PDP-11 Getting DOS On The Air
DEC-11-SYDD-D
October 1971

READER'S COMMENTS

Digital Equipment Corporation maintains a continuous effort to improve the quality and usefulness of its publications. To do this effectively we need user feedback -- your critical evaluation of this manual.

Please comment on this manual's completeness, accuracy, organization, usability and readability.

Did you find errors in this manual? If so, specify by page.

How can this manual be improved?

Other comments?

Please state your position. _____ Date: _____

Name: _____ Organization: _____

Street: _____ Department: _____

City: _____ State: _____ Zip or Country _____

Fold Here

Do Not Tear - Fold Here and Staple

FIRST CLASS
PERMIT NO. 33
MAYNARD, MASS.

BUSINESS REPLY MAIL
NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

Postage will be paid by:

digital

Digital Equipment Corporation
Software Information Services
146 Main Street, Bldg. 3-5
Maynard, Massachusetts 01754