CHAPTER 2

SYSTEM HARDWARE INSTALLATION OR MODIFICATION

2.1 INTRODUCTION

The previous chapter listed the hardware requirements of an ETOS system, the layouts of the ETOS materials within the memory and disk pack, and the ETOS support software. This chapter explains the installation of the materials contained in QUODATA ETOS distribution package and describes the procedures that transfer the ETOS software to the appropriate back-up medium for your unique hardware configuration.

QUODATA usually distributes ETOS on an RKØ5 disk pack or a System Industries removable pack. This distribution pack should not be used on a day-to-day basis as a production pack. It should only be used to create production packs. The procedures to create production packs are dependent upon your hardware configuration.

Installation procedures include placement of the ETOS board, modification of existing hardware to accommodate ETOS and tailoring of the ETOS software to your software and hardware specifications. This chapter will discuss the hardware modifications necessary to operate single user ETOS. Chapter 3 will examine the modification necessary to configure the software to your hardware configuration. After you have completed these two chapters, a one-user ETOS system will be ready for running. Chapter 4 will explain how to start ETOS and Chapter 5 will discuss the implementation of additional users.

2.2 BOOTING RKØ5 ETOS DISK PACKS

Before you can install ETOS, you must be able to boot stand-alone
OS/8 on the ETOS pack. These sections describe the procedures
appropriate for all types of installations.

The information in this section is used to boot the ETOS
distribution pack. The pack used for daily operation of ETOS
(the production pack) is a copy of the distribution pack tailored
for your site. Therefore, this information is also used to boot
the daily operational or production ETOS pack.

2.2.1 Booting With a Hardware Bootstrap

If you have an RKØ5 hardware bootstrap (MI8) for the PDP8E (or
PDP8M), lift the SW switch up and down to boot the disk. If you
have a KM8-AA bootstrap for the PDP8A set to the RKØ5 disk, lift
the BOOT switch up and down to boot the disk. If you do not have
a hardware bootstrap for your disk, use the appropriate
information from one of the following sections to boot OS/8.

2.2.2 Booting With a Front Panel

The following instructions explain the initiation of OS/8 from
the front panel. These instructions are valid for PDP8E (or
PDP8M) and PDP8A computers. The PDP8E and PDP8A designations
respectively for front panel keys are enclosed in parentheses.

  1) Insert the ETOS disk pack in drive Ø. Set the switches on
      the drive to RUN and WT PROT. Insure that the RUN, ONCYL
      and WTPROT lights are on. It takes approximately five
      seconds for all lights to come on.

  2) Lift or depress halt key and restore to original position
      (HALT or HLT).

  3) Set the switch register to ØØØØ.

4) Depress extended load address key (EXTD ADDR LOAD or LXA).

5) Set switch register to 0030.

6) Depress load address key (ADDR LOAD or LA).

7) Set switch register to 6743.

8) Lift or depress deposit key (DEP or D NEXT).

9) Set switch register to 5031.

10) Lift or depress deposit key (DEP or D NEXT).

11) Set switch register to 0030.

12) Depress load address key (ADDR LOAD or LA).

13) Depress the initialize key (CLEAR or INIT).

14) Depress the run key (CONT or RUN).

[handwritten marginal note: Adr 0030 / Load 6743 5031 / St. 0030]

The system responds with a ".", indicating that stand-alone OS/8 is up and running.

2.2.3 Booting From a Non-RK05 Peripheral

If you do not have a front panel but have a non-RK05 peripheral hardware bootstrap, use the following instructions to bootstrap the RK05.

1) Hardware bootstrap the non-RK05 peripheral. The non-RK05 peripheral must contain media with OS/8 on it. When bootstrapped, it produces a ".".

2) Enter the commands shown in Figure 2-1.

Figure 2-1
ODT Bootstrap for RK05 Disk

```
.ODT<RET>

27/ XXXX    6007<LF>
00030/ XXXX 6743<LF>
00031/ XXXX 5031<RET>
27G
```

The system responds with a ".", indicating that stand-alone OS/8 from the ETOS pack is up and running.

As an alternative to using ODT to type in the bootstrap, you can run the OS/8 BOOT command from your non-RK05 peripheral from the disc to initiate OS/8. To boot the RK05, enter the command contained in Figure 2-2 after bootstrapping a non-RK05 peripheral.

Figure 2-2
Using the OS/8 Command, BOOT, to Boot an RK05

.BOOT/RE (RE is the mnemonic for the RK05)
.

Stand-alone OS/8 is now up and running on the RK05.

2.2.4   Booting on a DEC Data System

If you have a DEC Data System, pressing the INIT key on the front of the desk causes the numbers "0123" to be printed. If you mount a COS system diskette in the left-hand floppy drive and press <RET>, you can run single-user COS. In the ETOS distribution kit, there is a floppy diskette labeled "RK05 BOOT DISK". If you mount this floppy in the left-hand drive after "0123" is printed and you press <RET>, the RK05 is automatically booted.

2.3 BOOTING SYSTEM INDUSTRIES (SI) ETOS DISK PACKS

Before you can install ETOS, you must be able to boot stand-alone OS/8 on the ETOS pack. These sections describe the procedures appropriate for all types of installations.

The information in this section is used to boot the ETOS distribution pack. ETOS is distributed on a removable SI disk pack. When ETOS runs, it executes on the fixed SI disk pack in port 0. To get ETOS running you must boot the distributed removable pack and copy it onto the fixed pack. Once the information is copied, utilize a fixed pack bootstrap to boot this production pack. The fixed pack bootstrap information starts in section 2.3.5.

2.3.1 Booting a Removable Pack with a Front Panel

The following instructions explain the initiation of OS/8 from the front panel. These instructions are valid for PDP8E (or PDP8M) and PDP8A computers. The PDP8E and PDP8A designations respectively for front panel keys are enclosed in parentheses.

1) Load the removable pack in the drive on port 0. Follow the procedures contained in the disk drive instruction manual for disk cartridge load. Set the RUN switch on drive 0 down so that the light is on. Set the PROT FIXED switch down so that the light is off. Set the PROT RMVBL switch up so that the light is on. Insure that the READY light is on. It takes approximately ninety seconds for this light to come on.

2) Lift or depress halt key and restore to original position (HALT or HLT).

3. Set switch register to 0400.

4. Depress load address key (ADDR LOAD or LA).

5. Depress extended load address key (EXTD ADDR LOAD or LXA).

6. Set switch register to 6502.

7. Lift or depress deposit key (DEP or D NEXT).

8. Set switch register to 1213.

9. Lift or depress deposit key (DEP or D NEXT).

10. Set switch register to 6517.

11. Lift or depress deposit key (DEP or D NEXT).

12. Set switch register to 7200.

13. Lift or depress deposit key (DEP or D NEXT).

14. Set switch register to 6512.

15. Lift or depress deposit key (DEP or D NEXT).

16. Set switch register to 7332.

17. Lift or depress deposit key (DEP or D NEXT).

18. Set switch register to 6514.

19. Lift or depress deposit key (DEP or D NEXT).

20. Set switch register to 7300.

21. Lift or depress deposit key (DEP or D NEXT).

22. Set switch register to 6501.

23. Lift or depress deposit key (DEP or D NEXT).

24. Set switch register to 5210.

25. Lift or depress deposit key (DEP or D NEXT).

26. Set switch register to 5005.

27. Lift or depress deposit key (DEP or D NEXT).

28. Set switch register to 0414.

29. Lift or depress deposit key (DEP or D NEXT).

30. Set switch register to 0400.

31. Lift or depress deposit key (DEP or D NEXT).

32. Set switch register to 0000.

33. Lift or depress deposit key (DEP or D NEXT).

34. Set switch register to 0400.

35. Depress load address key (ADDR LOAD or LA)

36. Depress the initialize key (CLEAR or INIT).

37. Depress the run key (CONT or RUN).

The system responds with a ".", indicating that stand-alone OS/8 is up and running. The above procedure is valid for S.I. 3040 controllers with a device code of 50/51. If you have a different device code, you must use existing procedures to boot the fixed disk and access the removable pack.

Now that the removable pack is booted, skip to 2.3.4 to copy the removable back to the fixed pack.

2.3.2 Booting a Removable Pack from a Non-SI Peripheral

If you do not have a front panel but you have a non-SI peripheral hardware bootstrap, use the following instructions to bootstrap the SI removable disk.

1) Hardware bootstrap the non-SI peripheral. The non-SI peripheral must contain media with OS/8 on it. When bootstrapped, it produces a ".".

2) Enter the commands shown in Figure 2-3.

Figure 2-3
ODT Bootstrap for S.I. Removable Disk

```
.ODT<RET>

377/    XXXX    6007<LF>
0400/   XXXX    6502<LF>
0401/   XXXX    1213<LF>
0402/   XXXX    6517<LF>
0403/   XXXX    7200<LF>
0404/   XXXX    6512<LF>
0405/   XXXX    7332<LF>
0406/   XXXX    6514<LF>
0407/   XXXX    7300<LF>
0410/   XXXX    6501<LF>
0411/   XXXX    5210<LF>
0412/   XXXX    5005<LF>
0413/   XXXX    0414<LF>
0414/   XXXX    0400<LF>
0415/   XXXX    0000<RET>
377G
```

The system responds with a ".", indicating that stand-alone OS/8 from the removable ETOS pack is up and running.

As an alternative to using ODT to type in the bootstrap, you can run the BOOT program to initiate OS/8. To boot the fixed disk, enter the command contained in Figure 2-4 after bootstrapping a non-System Industries peripheral.

Figure 2-4
Using the OS/8 Command, BOOT, to Bootstrap the SI Removable Disk

```
.BOOT/ST
.
```

Stand-alone OS/8 is now up and running on the System Industries disk.

Note: The version of BOOT.SV must have been copied to the non-System Industries OS/8 peripheral from the System Industries pack prior to the execution of the above command. The stand-alone OS/8 boot program does not contain the "/ST" option.

2.3.3 Booting a Removable Pack on a DEC Data System

If you have a DEC Data System, pressing the INIT key on the front of the desk causes the number "0123" to be printed. If you mount a COS system diskette in the left-hand floppy drive and press <RET>, you can run single-user COS. In the ETOS distribution kit, there is a floppy diskette labeled "SI REMOVABLE BOOT DISK". If you mount this floppy in the left-hand drive after "0123" is printed and your press <RET>, the SI removable disk is automatically booted.

2.3.4 Creating an ETOS Fixed Pack

ETOS runs from the fixed pack in port 0. Therefore, you must copy the contents of the removable pack onto the fixed pack before you can run ETOS. The fixed pack must be formatted before this copy may be accomplished (see 2.4.2). Use the procedure contained in Figure 2-5 to copy the removable pack to the fixed one.

Note: SICOPY, for System Industry copy is a program on the removable pack.

Figure 2-5
Copying the Removable Disk to the Fixed Disk

.R SICOPY

'SICOPY' V1.021
SOURCE PORT (0-3)?0 SOURCE PLATTER (F OR R)? R
DESTINATION PORT (0-3)?0 DESTINATION PLATTER (F OR R)? F
ARE YOU READY? Y

DONE
    .

When running the SICOPY program, you may get an error in the format illustrated in Figure 2-6.

Figure 2-6
SICOPY Error Format

```
READ ERROR (OR WRITE ERROR)
CONTROL REGISTER         XXXX
STATUS REGISTER          XXXX
SEEK STATUS REGISTER     XXXX
TRACK                    XXXX
```

The program retries the copy of the track in error and if it is successful, the copy resumes. If the program resumes, ignore the error. If the copy of the track in error is retried eight times, the program stops with the message "HARD ERROR, ABORTING". If this error occurs, there is an error in set-up or a serious hardware problem. After the program successfully completes, the only step left in creating a fixed ETOS pack is to copy the correct OS/8 system area to the fixed pack. .

To create a fixed pack OS8 system area, use the procedure of Figure 2-7.

Figure 2-7
Creating a Fixed Pack OS/8 System Area

```
.R BUILD
$BOOT
SYS BUILT

 .
```

You now have a fixed ETOS pack with stand-alone OS/8. Remove the distribution pack and deposit it in a safe place. This pack should never be used for any purpose other than restoration of the fixed ETOS pack.

2.3.5 Booting a Fixed Pack on a PDP8E Processor, Using the Front
        Panel

To ETOS on a daily basis, you must boot the fixed disk. This
procedure should be followed each time the system is brought up
after being powered down.

The timing of the PDP8A processor is different from the PDP8E
processor. Therefore, there are two different bootstraps. The
following instructions explain the initiation of OS/8 from the
front panel on a machine with a PDP8E processor. It is possible
to have a PDP8E processor in a PDP8A box (8A600 or 8A620).
Therefore, the PDP8E and PDP8A designations respectively, for
front panel keys are enclosed in parentheses.

   1. Insert a removable pack in the drive on port 0. Even though
       you are booting the fixed disk, you must have a removable
       pack in the drive or it will not come up and running. If
       you have no particular removable pack which you wish to
       access, you may insert a blank pack. Set the RUN switch on
       that drive down so that the light is on. Set the PROT FIXED
       switch up so that the light is on. Set the PROT RMVBL
       switch up so that the light is on. Insure that the READY
       light is on. It takes approximately ninety seconds for this
       light to come on.

   2. Lift or depress halt key and restore to original position
       (HALT or HLT).

   3. Set switch register to 0000.

   4. Depress load address key (ADDR LOAD or LA).

   5. Depress extended load address key (EXTD ADDR LOAD or LXA).

   6. Set switch register to 6502.

   7. Lift or depress deposit key (DEP or D NEXT).

   8. Set switch register to 0000.

   9. Lift or depress deposit key (DEP or D NEXT).

   10. Set switch register to 6517.

11.   Lift or depress deposit key (DEP or D NEXT).

12.   Set switch register to 6512.

13.   Lift or depress deposit key (DEP or D NEXT).

14.   Set switch register to 6514.

15.   Lift or depress deposit key (DEP or D NEXT).

16.   Set switch register to 5005.

17.   Lift or depress deposit key (DEP or D NEXT).

18.   Set switch register to 0000.

19.   Depress load address key (ADDR LOAD or LA).

20.   Depress the initialize key (CLEAR or INIT).

21.   Depress the run key (CONT or RUN).

The system responds with a ".", indicating that stand-alone OS/8 is now up and running.

2.3.6   Booting a Fixed Pack on a PDP8A Processor, Using the Front Panel

The procedure contained in 2.3.5 is used to boot a System Industries disk on a system with a PDP8E processor. The following procedures are used to boot a System Industries fixed disk on a system with a PDP8A processor.

1.   Insert a removable pack in the drive on port 0. Even though you are booting the fixed disk, you must have a removable pack in the drive or it will not come up and running. If you have no particular removable pack which you wish to access, you may insert a blank pack. Set the RUN switch on the drive down so that the light is on. Set the PROT FIXED switch is up so that the light is on. Set the PROT RMVBL switch up so that the light is on. Insure that the READY light is on. it takes approximately ninety seconds for this light to come on.

2.   Depress halt (HLT) key.

3. Set switch register to 0400.

4. Depress load address (LA) key.

5. Depress extended load address (LXA) key.

6. Set switch register to 6502.

7. Lift or depress deposit (D NEXT) key.

8. Set switch register to 1211.

9. Lift or depress deposit (D NEXT) key.

10. Set switch register to 6517.

11. Lift or depress deposit (D NEXT) key.

12. Set switch register to 7200.

13. Lift or depress deposit (D NEXT) key.

14. Set switch register to 6512.

15. Lift or depress deposit (D NEXT) key.

16. Set switch register to 6514.

17. Lift or depress deposit (D NEXT) key.

18. Set switch register to 6501.

19. Lift or depress deposit (D NEXT) key.

20. Set switch register to 5206.

21. Lift or depress deposit (D NEXT) key.

22. Set switch register to 5005.

23. Lift or depress deposit (D NEXT) key.

24. Set switch register to 0412.

25. Lift or depress deposit (D NEXT) key.

26. Set switch register to 0400.

27. Lift or depress deposit (D NEXT) key.

28. Set switch register to 0000.

29. Lift or depress deposit (D NEXT) key.

30. Set switch register to 0400.

31. Depress load address (LA) key.

32. Depress the initialize (INIT) key.

33. Depress the run (RUN) key.

The system responds with a ".", indicating that stand-alone OS/8 is now up and running.

2.3.7 Booting a Fixed Pack From a Non-SI Peripheral

If you do not have a front panel but you have a non-SI peripheral hardware bootstrap, use the following instructions to bootstrap the SI fixed pack. These instructions are valid for PDP8E (or PDP8M) or PDP8A computers.

1) Hardware bootstrap the non-SI peripheral. The non-SI peripheral must contain media with OS/8 on it. When bootstrapped, it produces a ".".

2) Enter the commands shown in Figure 2-8.

Figure 2-8
ODT Bootstrap for the S.I. Fixed Disk

```
.ODT<RET>

377/    XXXX    6007<LF>
0400/   XXXX    6502<LF>
0401/   XXXX    1211<LF>
0402/   XXXX    6517<LF>
0403/   XXXX    7200<LF>
0404/   XXXX    6512<LF>
0405/   XXXX    6514<LF>
0406/   XXXX    6501<LF>
0407/   XXXX    5206<LF>
0410/   XXXX    5005<LF>
0411/   XXXX    0412<LF>
0412/   XXXX    0400<LF>
0413/   XXXX    0000<LF>
377G
```

The system responds with a ".", indicating that stand-alone OS/8 is now up and running.

As an alternative to using ODT to type in the bootstrap, you can run the BOOT program to initiate OS/8. To boot the fixed disk, enter the command contained in Figure 2-9 after bootstrapping a non-System Industries peripheral.

Figure 2-9
Using the OS/8 Command, BOOT, to Bootstrap the SI Fixed Disk

```
.BOOT/SB        (SB is mnemonic for the S.I. Fixed Disk)
.
```

Stand-alone OS/8 is now up and running on the System Industries disk.

Note: The version of BOOT.SV must have been copied to the OS/8 peripheral from the System Industries pack prior to the execution of the above command. The stand-alone OS/8 BOOT program does not contain the "/SB" option.

2.3.8 Booting a Fixed Pack on a DEC Data System

If you have a DEC Data System, pressing the INIT key on the front
of the desk causes the number "0123" to be printed. If you mount
a COS system diskette in the left-hand floppy drive and press
<RET>, you can run single-user COS. In the ETOS distribution
kit, there is a floppy diskette labeled "SI FIXED BOOT DISK". If
you mount this floppy in the left-hand drive after "0123" is
printed and you press <RET>, the SI fixed disk is automatically
booted.

2.4 FORMATTING A DISK PACK

When a disk is sent from QUODATA Corporation or one of the media
suppliers, it is not ready for use. The disk needs to have track
identification marks written on it so that the software can
properly access the disk. This process is known as formatting.
Before you can make a copy of the ETOS distribution pack, you
must have a formatted pack to copy onto. The following sections
illustrate the procedures necessary to format RK05 and System
Industries disk packs.

2.4.1 Formatting An RK05 Disk Pack

To format an RK05 disk, you must first boot an OS/8 mass storage
medium which contains the RKLFMT program. The ETOS distribution
pack can be booted in RK05 drive 0, since the RKLFMT program is
on that pack. The RKLFMT program is used to format RK05 disks.
This program may be used instead of the COS program RKEMRK or the
OS/8 V3A program RKEFMT. Additional documentation on RKLFMT is
provided in Appendix K, OS/8 Handbook Update.

To format an RK05 disk, utilize the following procedure.

 1) Boot an OS/8 mass storage medium which contains the RKLFMT

program. The ETOS distribution pack may be used.

2) Enter the command R RKLFMT <RET>.

3) The program responds with an identification header and then prints "FORMAT DISK 0?". At this time, insert the disk or disks you want formatted in drive 0, 1, 2 or 3. Even if you initiated the RKLFMT program from a disk in drive 0, you can remove that disk at this time. If you do not remove a disk which contains valid information, insure that it is write protected. When you have set up all drives, answer the initial question with a Y or an N.

4) Answer the questions "FORMAT DISK n?" where n varies from 1 to 7. Since ETOS supports a maximum of four drives, the answer to the question will always be N when the disk drive is 4 through 7. The answer to the other questions depends on which drives contain disks ready to be formatted. If a drive does not exist, answer N for the question asking if you want to format that drive.

5) The system asks "ARE YOU SURE?" If you have made a mistake in answering any of the initial eight questions, enter N. The format questions are asked again. If you have made no mistakes, enter Y. Before you enter Y, insure that all drives, containing disks you want to format, are set to RUN and that the RDY and ON CYL lights are on.

6) When all formatting is complete, the system prints a completion message and then prints "FORMAT SAME DISK(S) AGAIN?" Remove all disks which have just been formatted. If you do not wish to format more disks, insure that an OS/8 system pack is mounted in drive 0, write enabled and enter ^C. If you have more disks to format, insert them in the proper drives. If the set-up of the drives is the same as the previous set-up, enter Y. If the set-up is different, enter N. The eight formatting questions are asked again.

Figure 2-10 contains a sample run of the RKLFMT program.

Figure 2-10
Sample Run of RKLFMT

```
.R RKLFMT
RK8E/RK8L DISK FORMATTER PROGRAM
FOR ALL QUESTIONS, ANSWER Y FOR YES OR N FOR NO
FORMAT DISK 0? N
FORMAT DISK 1? Y
FORMAT DISK 2? Y
FORMAT DISK 3? N
FORMAT DISK 4? N
FORMAT DISK 5? N
FORMAT DISK 6? N
FORMAT DISK 7? N
ARE YOU SURE? Y

RK8E/RK8L DISK FORMATTER PASS COMPLETE
FORMAT SAME DISK(S) AGAIN? Y

RK8E/RK8L DISK FORMATTER PASS COMPLETE
FORMAT SAME DISK(S) AGAIN? ^C
```

.

2.4.2   Formatting a System Industries Disk Pack

To format a System Industries disk, boot an OS/8 mass storage
medium which contains the 3040 program.   The ETOS distribution
pack or the fixed disk you created, can be booted in port 0
(using the procedures of 2.3) since the 3040 program is on those
packs.   The 3040 program is used to diagnose disk errors (see
8.4.2) and format disks.   This program is a core image copy of
the standard paper tape distributed by System Industries.
Additional documentation on 3040.SV is provided in the System
Industries 30/40 Controller Manual.

To format a System Industries disk, utilize the following
procedure.

1)   Boot an OS/8 mass storage medium which contains the 3040
     program.

2)   Enter the command R 3040.

3) The system responds with "ENTER DISK TEST INFORMATION". At
   this time, insert the disk or disks you want formatted in
   port 0 or 1. Even if you initiated the 3040 program from
   the removable pack in port 0, you can remove that disk at
   this time. If you do not remove a disk which contains valid
   information, insure that it is write protected. Insure that
   the READY light is on for all ports which you want to use.
   When you have set up all fixed and removable packs, answer
   the question 00XY, where "Y" is the port number plus four
   and "X" specifies whether you want to format the removable
   pack, the fixed pack or both packs on this port. You don't
   need to press carriage return after you specify your answer.
   If "X" equals "3", the system formats the fixed disk. If
   "X" equals "5", the system formats the removable disk. If
   "X" equals "7", the system formats both fixed and removable
   disks. For example, if you want to format the fixed pack in
   port 1, you would specify 0035.

   The system continues to request disk test information until
   all ports, which you wish to utilize, have been specified.
   When all ports have been specified, enter 0000.

4) The system prints "ENTER UNIT ADDRESS". Enter the device
   code of the system industry disk. This will always be
   "0050" unless you have a non-standard disk.

5) The system prints "ENTER FORMAT LOOP CONTROL". Enter 0000.

6) The system prints "ENABLE FORMAT SWITCH, ERROR AND TYPE OUT
   CONTROLS". Set the switch on the back of all ports you are
   using to FORMAT. Before you enter any specification, insure
   that all ports, containing disks you want to format, are set
   to run and that the READY light is on. Enter 0040 to
   initiate formatting.

7) When all formatting is complete, the system prints "FORMAT
   COMPLETE" and jumps to location 07600. If you have an OS/8
   system pack in port 0, the system returns to the OS/8
   monitor. If you do not have an OS/8 system pack in drive 0,
   the system hangs and you must reBOOT the system.

A sample run of the 3040 program is shown in Figure 2-11.

Figure 2-11
Sample Run of 3040

```
.R 3040

ENTER DISK TEST INFORMATION 0074      (port  0,  fixed  and
                                       removable)
ENTER DISK TEST INFORMATION 0035      (port 1, fixed)
ENTER DISK TEST INFORMATION 0000
ENTER UNIT ADDRESS 0050
ENTER FORMAT LOOP CONTROL 0000
ENABLE FORMAT SWITCH, ERROR AND TYPE OUT CONTROLS 0040

FORMAT COMPLETE
.
```

2.5 COPYING ETOS PACKS

The distribution pack from QUODATA should never be used as a
production pack. It should be inserted in a disk drive only to
copy it onto another disk pack or onto some other media as
dictated by your system configuration. The distribution pack
should then be placed in a secure place and used only to
construct another production pack if the current one becomes
corrupted. The backup philosophy is detailed in 8.12.

If you have a System Industries drive, you have already copied
the distribution pack onto the fixed pack in port 0 using the
procedures in 2.3.4. This section can be used to copy this fixed
pack. The documentation on copying packs is also used to copy
non-system disks. These non-system disks may be standard OS/8 or
COS format disks or special ETOS packs which act as an extension
of the system pack (see 5.11). If you have two or more RK05
drives, utilize 2.5.1 to perform an image copy of one RK05 pack
to another RK05 pack. If you have one or two System Industries
drives, utilize 2.5.2 to perform an image copy of one fixed or
removable pack to another fixed or removable pack. If you have
one RK05 drive, utilize 2.5.3 to copy an RK05 disk pack to
another type of OS/8 supported mass storage.

2.5.1 Copying an RK05 Disk Pack

To copy an RK05 disk, you must first boot an OS/8 mass storage medium which contains the RKCOPY program such as the QUODATA ETOS distribution pack. The ETOS distribution pack can be booted in RK05 drive 0, since the RKCOPY program is used to copy RK05 disks. This program may be used instead of the COS program PIP or the OS/8 program FOTP.

To copy an RK05 disk, use the following procedure.

1) Boot an OS/8 mass storage medium which contains the RKCOPY program.

2) Enter the command R RKCOPY <RET>.

3) The system responds with a program header and then prints "FROM DRIVE?". At this time, insert the disk you want copied in one of your RK05 drives. Set the drive to RUN and insure that it is WRITE PROTECTED. Insert the disk you want to copy onto, in another one of your RK05 drives. Set the drive to RUN and insure that it is WRITE ENABLED. This disk must be formatted or a write error occurs. When you have set up the two drives, answer the initial question with a 0, 1, 2, or 3, depending upon which drive is the source.

4) The system prints "TO DRIVE?". Enter the drive number which contains the disk you are copying onto. Any existing data stored on this disk is overwritten.

5) The system prints "STRIKE ANY CHARACTER TO START COPY:". When you are confident that your set-up is correct, hit any key on the console terminal to initiate the copy.

If you have made an error in the drive specifications, enter ^C and run the program again.

6) The copy process takes approximately thirty seconds. When the copy is complete, the system prints "MONITOR REMOUNTED?" Remove the disks used for copying at this time. If you have just copied the ETOS distribution pack, store the original pack in a safe place. You must insert a disk, which contains an OS/8 system head, into drive 0. The copy of the distribution pack copy, which is known as the production pack, contains an OS/8 monitor. When you have inserted a

disk which contains OS/8 in drive 0, enter Y to return to
the OS/8 monitor. If you have more disks to copy, run the
RKCOPY program again using the production pack in place of
the distribution pack.

A sample run of the RKCOPY program which illustrates the back-up
of the ETOS distribution pack is shown in Figure 2-12.

Figure 2-12
Sample Run of RKCOPY

```
.R RKCOPY
'RK05' V3.005
FROM DRIVE? 0
TO DRIVE? 1
STRIKE ANY CHARACTER TO START COPY: <RET>

IS MONITOR REMOUNTED? Y
.
```


2.5.2 Copying a System Industries Disk Pack

To copy a System Industries disk, you must boot an OS/8 mass
storage medium which contains the SICOPY program. The ETOS
distribution pack or the fixed disk you created using the
procedures of 2.3.4, can be booted in port 0, since the SICOPY
program is on these packs. The SICOPY program is used to copy
System Industries packs. If you want to make a copy of the
production pack, copy the fixed disk in port 0 to a removable
pack in either port.

To copy a System Industries disk, use the following procedure.

 1) Boot an OS/8 mass storage medium which contains the SICOPY
     program. (see 2.3)

 2) Enter the command R SICOPY <RET>.

 3) The system responds with "SOURCE PORT (0-3)"? At this time,
     insert the disk or disks you want copied in a drive. If you
     are copying a fixed pack, the pack will always be inserted.
     Set the drive to RUN and insure that it is write protected.
     Insert the disk you want to copy onto in a drive. If the

disk you are copying onto is a fixed pack, the pack will always be inserted. Set the drive to RUN and insure that it is write enabled. This disk must be formatted as a write error occurs. Even if you initiates the SICOPY program from a removable disk in drive 0, you can remove that disk now. When you have set up both packs, answer the initial question with the port number (0 or 1), which contains the disk you are copying.

4) The system prints "SOURCE PLATTER (F or R)?" Enter <u>F</u> if the disk you are copying is a fixed pack. Enter <u>R</u> if the disk you are copying is a removable pack.

5) The system prints "DESTINATION PORT (0-3)?" Enter the port number (0 or 1), which contains the disk you are copying onto. Any existing data stored on this pack is overwritten.

6) the system prints "DESTINATION PLATTER (F or R)?". Enter <u>F</u> if the disk you are copying onto is a fixed pack. Enter <u>R</u> if the disk you are copying is a removable pack.

7) The system prints "ARE YOU READY?". When you are confident that your set-up is correct, hit any key on the console character to initiate the copy. If you have made an error in the drive specifications, enter <u>^C</u> before hitting the <RET> and run the program again.

8) The copy process takes approximately two and a half minutes. When the copy is complete, the system prints "DONE" and returns to the OS/8 monitor. The medium which the SICOPY program was initiated from must contain an OS/8 system head. If it does not, the system hangs after printing "DONE" and you must boot an OS/8 medium from scratch.

When running the SICOPY program, you may get the error in the format illustrated in Figure 2-13.

<div align="center">

Figure 2-13
SICOPY Error Format
</div>

```
READ ERROR (or WRITE ERROR)
CONTROL REGISTER          XXXX
STATUS REGISTER           XXXX
SEEK STATUS REGISTER      XXXX
TRACK                     XXXX
```

The program retries the copy of the track in error and if it is successful, the copy resumes. If the program resumes, ignore the error. If the copy of the track in error is retried eight times,

the program stops with the message "HARD ERROR, ABORTING". If
this error occurs, there is a problem in set-up or a serious
hardware problem.

A sample run of the SICOPY program which illustrates the back-up
of the fixed ETOS production pack is shown in Figure 2-14.

Figure 2-14
Sample Run of SICOPY

```
.R SICOPY
'SICOPY' V1.021
SOURCE PORT (0-3)? 0 SOURCE PLATTER (F or R)? F
DESTINATION PORT (0-3)? 0 DESTINATION PLATTER (0-3)? R
ARE YOU READY? Y

DONE
.
```

2.5.3 Copying the ETOS Distribution Pack onto a Non-Disk Medium

FOTP may be used to backup the stand-alone portion of a pack
under single user OS/8. This program may also be used to backup
an ETOS account under ETOS, if the peripheral is supported under
ETOS time sharing. However, there is no way to back the ETOS
portion of a pack onto a medium not supported under ETOS. Also,
backup under ETOS must be done one account at a time, which is
tedious. QUODATA's BACKUP program which runs under stand-alone
OS/8, eliminates this problem. Backup copies one or more volumes
of one media to one or more volumes of another media. To run
BACKUP, use the procedure given in Figure 2-15.

<div align="center">

Figure 2-15
BACKUP Dialogue
</div>

```
.R BACKUP
'BACKUP' V2.005
*output device:<input device:
STARTING INPUT BLOCK? inblock
STARTING OUTPUT BLOCK? outblock
NUMBER OF BLOCKS? numblock
*
```

You first specify the input and output device. These devices can be any OS/8 device which is built into the stand-alone system. Use BUILD to build in any handler not contained in the system. The starting input block is the beginning block for transfer on the input device. The starting output block is the beginning block for transfer on the output device. The number of blocks is the number of 256 word records to transfer.

If the number of blocks transferred is greater than the size of the output device, the system displays "MOUNT NEXT OUTPUT DEVICE:" as each output medium is exhausted. Therefore, you can copy a disk to ten DECtapes in this manner.

If the number of blocks transferred is greater than the size of the input device, the system displays "MOUNT NEXT INPUT DEVICE:" as each input medium is exhausted. Mount the next input medium and press carriage return. You can copy ten DECtapes to a disk in this manner.

When the program has completed the transfer, it responds with a "*". Enter ^C to exit. The backup media is not OS/8 file structured and no directory manipulation is performed. This program performs only block transfers. File oriented transfers should be performed with PIP or FOTP.

Stand-alone OS/8 cannot handle a device which is greater than 7777 (base 8) or 4095 (base 10) blocks. To use an RK05, OS/8

treats it as two devices of 6260 (base 8) or 3248 (base 10) blocks, instead of one device of 14540 (base 8) or 6496 (base 10) blocks. To use a System Industries 30/40 disk, OS/8 treats it as four devices of 6260 (base 8) or 3248 (base 10) blocks, instead of one device of 31400 (base 8) or 13056 (base 10) blocks. Note that 3248 multiplied by 4 is 12992. In order to maintain compatability with RK05 systems, these device areas are kept at 3248 blocks. There is room for 3264 blocks, but these extra 16 blocks are not used. Therefore, under OS/8, 64 blocks on the pack are unused.

Since OS/8 treats the disks as multiple devices, you must run the BACKUP program more than once, to back up an entire disk. The dialogue of Figure 2-16 copies an entire RK05 disk to ten DECtapes. The dialogue of Figure 2-17 copies the ten DECtapes back to an RK05 disk.

The dialogue of Figure 2-18 may be used to copy a System Industries disk to eight RK05 packs. Note that the length of each area of the pack is specified as 6300 (base 8) or 3264 (base 10) blocks, not 6260 (base 8) or 3248 (base 10) blocks which are used under stand-alone OS/8 processing. This is to insure that all data is transferred to the backup medium from the ETOS pack, with no gaps.

```
                        Figure 2-16
              Copying an RK05 Disk to Ten DECtapes

.R BACKUP
'BACKUP' V2.005
*DTA0:<SYS:
STARTING INPUT BLOCK? 0
STARING OUTPUT BLOCK? 0
NUMBER OF BLOCKS? 6260
MOUNT NEXT OUTPUT DEVICE:<RET>
MOUNT NEXT OUTPUT DEVICE:<RET>
MOUNT NEXT OUTPUT DEVICE:<RET>
MOUNT NEXT OUTPUT DEVICE:<RET>
*DTA0:<RKB0:
STARTING INPUT BLOCK? 0
STARTING OUTPUT BLOCK? 0
NUMBER OF BLOCKS? 6260
MOUNT NEXT OUTPUT DEVICE:<RET>
MOUNT NEXT OUTPUT DEVICE:<RET>
MOUNT NEXT OUTPUT DEVICE:<RET>
MOUNT NEXT OUTPUT DEVICE:<RET>
*^C
.


                        Figure 2-17
              Copying Ten DECtapes to an RK05 Disk

.R BACKUP
'BACKUP' V2.005
*SYS:<DTA0:
STARTING INPUT BLOCK? 0
STARTING OUTPUT BLOCK? 0
NUMBER OF BLOCKS? 6260
MOUNT NEXT INPUT DEVICE:<RET>
MOUNT NEXT INPUT DEVICE:<RET>
MOUNT NEXT INPUT DEVICE:<RET>
MOUNT NEXT INPUT DEVICE:<RET>
*RKB0:<DTA0:
STARTING INPUT BLOCK? 0
STARTING OUTPUT BLOOCK? 0
NUMBER OF BLOCKS? 6260
MOUNT NEXT INPUT DEVICE:<RET>
MOUNT NEXT INPUT DEVICE:<RET>
MOUNT NEXT INPUT DEVICE:<RET>
MOUNT NEXT INPUT DEVICE:<RET>
*^C
.
```

Figure 2-18
Copying an SI Disk to Eight RK05 Disks

```
.R BACKUP
'BACKUP' V2.005
*RKA0:<SYS:
STARTING INPUT BLOCK? 0
STARTING OUTPUT BLOCK? 0
NUMBER OF BLOCKS? 6300
MOUNT NEXT OUTPUT DEVICE:<RET>
*RKA0:<DSK1:
STARTING INPUT BLOCK? 0
STARTING OUTPUT BLOCK? 0
NUMBER OF BLOCKS? 6300
MOUNT NEXT OUTPUT DEVICE:<RET>
*RKA0:<DSK2:
STARTING INPUT BLOCK? 0
STARTING OUTPUT BLOCK? 0
MOUNT NEXT OUTPUT DEVICE:<RET>
*RKA0:<DSK3:
STARTING INPUT BLOCK? 0
STARTING OUTPUT BLOCK? 0
MOUNT NEXT OUTPUT DEVICE:<RET>
*^C
.
```

2.6   BUILDING AN ETOS PACK FROM A NON-DISK DISTRIBUTION KIT

This section describes the process of building an ETOS
distribution pack from a peripheral other than an RK05 or System
Industries disk drive.   This section should only be used by
customers who have ETOS distribution on a non-disk medium (e.g.,
DECtape, floppy disk).

The following write-up assumes that you are building ETOS onto an
RK05 formatted pack or the fixed System Industries pack in port
0.   Any programs which are contained on the disk will be deleted.
"dev:" refers to the back-up medium (DTA0:, CSA0:, RXA0, etc.).

1A) If you have an RK05, mount a formatted pack in drive 0. Set
     the disk to RUN and make sure it's write-enabled.

1B) If you have a System Industries drive, mount a formatted
     removable pack in port 0. Set the disk to RUN and make sure
     it's write-enabled.

2) Mount the distribution medium marked "ETOS STAND-ALONE OS/8" on the appropriate peripheral. Bootstrap this medium, using the standard OS/8 bootstrap associated with the peripheral.

3) Enter the commands contained in Figure 2-19.

Figure 2-19
Building an ETOS Pack from a Non-Disk Medium

.RUN dev:BUILD
$PRINT

                  BUILD prints out all active device handlers. If you have a device on your system which is not active, enable it at this time. See 2-34, OS/8 Handbook for documentation on enabling inactive devices.

                  When the proper device handlers have been activated, enter

$BOOT
WRITE ZERO DIRECTORY? Y
.SAVE dev:BUILD

$ You are now running OS/8 off of the disk pack. Enter

.RUN dev:PIP
SYS:<dev:/S$ (= escape or alt mode)
.

This sequence of commands builds an OS/8 system, which has been modified for ETOS, onto the disk pack. All OS/8 programs which are to be used in stand-alone mode are copied onto the pack. These are unmodified OS/8 CUSPs except for PIP and RESORC. Also, the logical size of the disk pack is modified, since the ETOS pack needs the remaining area for virtual OS/8, swap tracks, user scratch blocks, user file storage, etc. This disk pack is identical to an ETOS distribution pack, except that the informational files are not on the pack and the virtual OS/8 programs have not been copied onto SYS. The informational files, which are discussed in 5.16 are not needed for daily operation of ETOS. They are contained on the media marked "INFORMATIONAL

FILES #X" and may be accessed at any time. Virtual OS/8 programs must be copied onto SYS before OS/8 can be run under ETOS. This procedure is followed after the configuration process.

2.7 ENABLING TIME SHARING

ETOS uses a special mode of the processor known as user mode. When the processor is in this mode, all Input/Output instructions (IOTs), change field instructions (CIFs), switch register instructions (OSR) and halt (HLT) instructions are trapped back to the monitor. If this were not the case, a user who executed a HLT (7402) would halt the real machine. This mode allows ETOS to give each user a separate virtual machine.

There are two hardware modules which allow this timesharing mode. Any machine which contains over 4K words of memory must have one of these two modules.

A PDP-8/E, /M or /F utilizes the KM8-E Memory Extension and Timeshare option. This M837 module includes a jumper which must be removed to allow ETOS to operate properly. The arrow on Figure 2-20 illustrates the location of this jumper.

A PDP-8/A utilizes either the KM8-E option or the KM8-AA I/O Option Board #2. For the KM8-3 option, the procedures of removing the jumper on the M837 module were described in the previous paragraph. To enable timesharing with the KM8-AA I/O Option Board of the PDP8/A, a switch on the M8317 module must be set to the OFF position. This switch is designated S2-1. It is the top switch in the lower of two groups of switches, located at the upper left-hand corner of the hex module. See Figure 2-21.

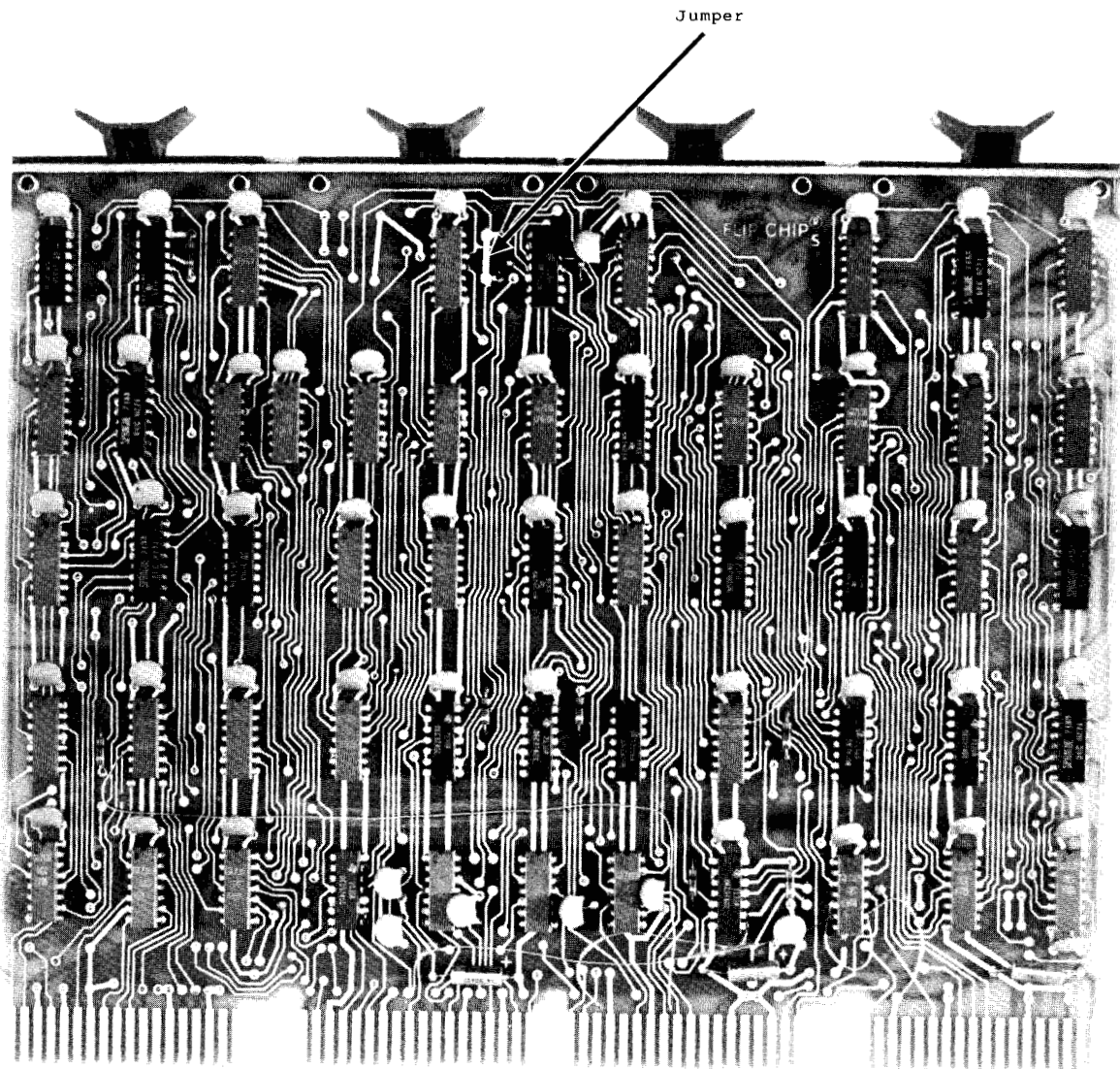Figure 2-20
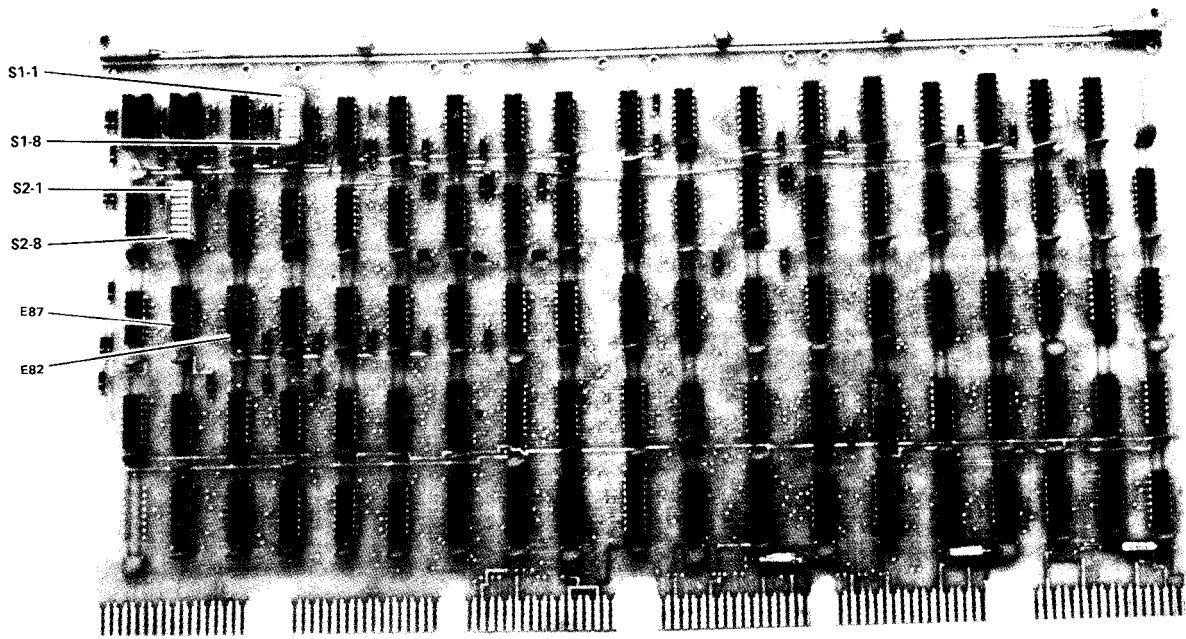The M837 Module Showing Location of Jumper (arrow)

Jumper

Figure 2-21
The M8317 Module with S2-1 Switch Indicated

S1-1

S1-8

S2-1

S2-8

E87

E82

7299-1

2.8 ENABLING THE CLOCK

When there is more than one user on the system, ETOS must schedule these users. Each job is run for a period of time (quantum) before another job is scheduled. The default quantum when the system is delivered is one hundred milliseconds. A clock module is required so that ETOS can perform this scheduling. In addition to installing the clock module, you must eventually enable the proper software module via the CONFIG program (see 3.3).

A PDP-8/E, /M or /F utilizes the DK8-EA (M882) line frequency clock, DK8-EC (M883 or M8830) crystal clock or DK8-EP (M860 and M518) programmable clock. One of these clock modules must be plugged into the omnibus. In addition, the DK8-EA module is connected to the J2 output on the back of the computer via a three wire cable.

The power fail/auto restart board (M848) also connects to the J2 output. Since ETOS does not utilize this board, you can remove it from the system. If you have a DK8-EA and an M848 board and you wish to have them both plugged in, use a Y cable to plug both modules into J2.
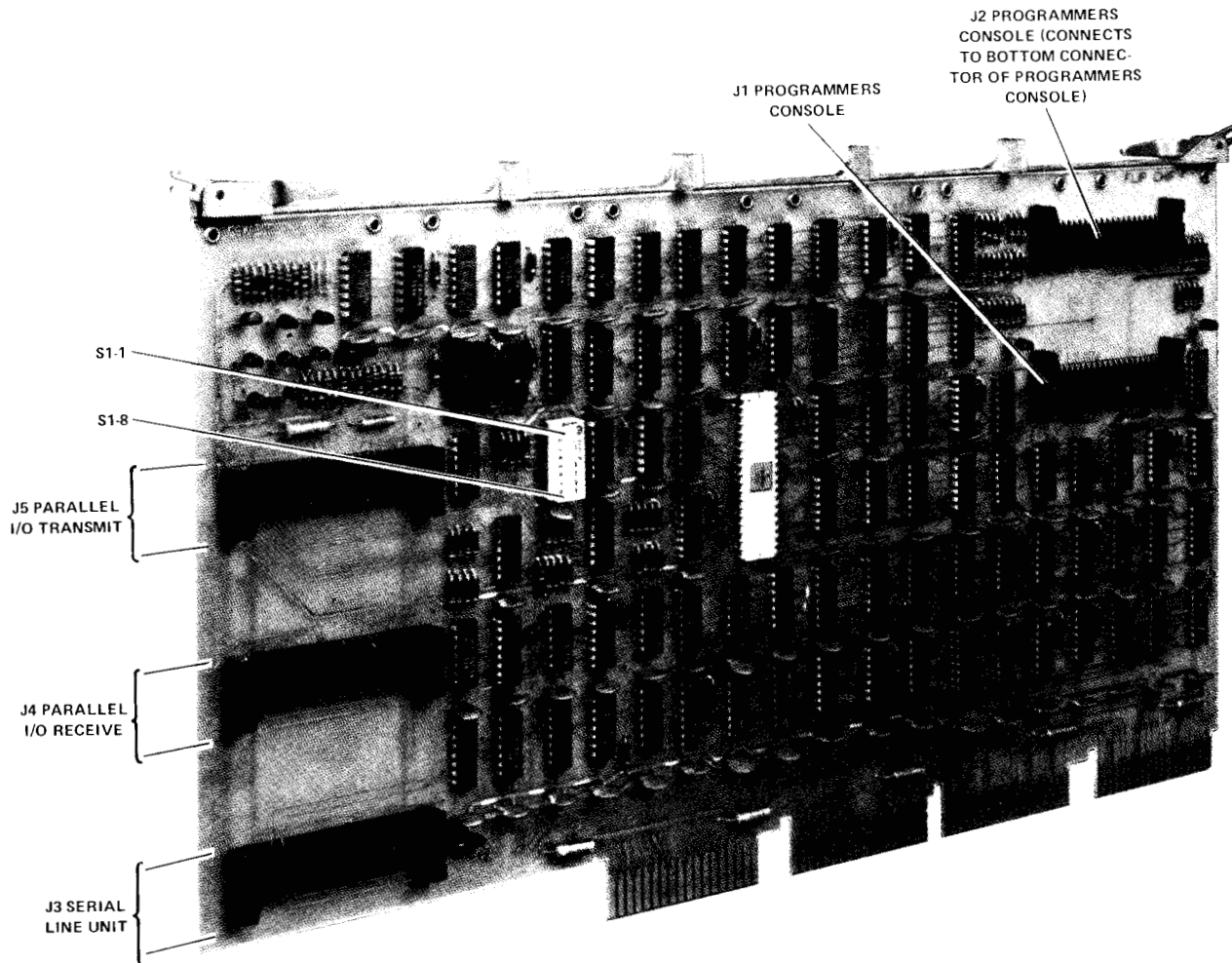
A PDP-8/A utilizes the DKC8-AA I/O Option Board #1. This M8316 module includes a switch which must be set to the ON positon to enable the clock. This switch designation is S1-5 on both the C etch and D etch versions of the M8316 board. Refer to Figures 2-22 or 2-23 for the location of this switch.

2.9 ENABLING THE LA8-A LINE PRINTER

An LA8-A line printer option consists of an LA180 printer connected to the DKC8-AA I/O Option Board #1 via a BC805A cable. The P1 end of the BC805S cable is inserted in the J5 output of the DKC8AA (M8316) module. See Figure 2-22 or 2-23 for the location of J5.

To enable the LA8-A printer, two procedures must be followed after the hardware is installed. Eventually, you must select the LA180D option in CONFIG (see 3.4). You must also set a switch on the DKC8-AA (M8316) module to the OFF position. There are two versions of the M8316 I/O Option Board: C etch and D etch. On the C etch DKC8-AA Board the switch is S1-4 (see Figure 2-22). On the D etch DCK8-AA Board, the switch is S1-9, (see Figure 2-23).

J2 PROGRAMMERS
CONSOLE (CONNECTS
TO BOTTOM CONNEC-
TOR OF PROGRAMMERS
CONSOLE)

J1 PROGRAMMERS
CONSOLE

S1-1

S1-8

J5 PARALLEL
I/O TRANSMIT

J4 PARALLEL
I/O RECEIVE

J3 SERIAL
LINE UNIT

7015 6

Figure 2-23
The D etch DKC8-AA I/O Board With Switches
S1-5 and S1-9 Indicated

REVISION LETTER
ON UNDERSIDE
OF BOARD (SIDE 2)

J5
PARALLEL I/O
TRANSMIT

SIDE 1

J2 PROGRAMMER'S
CONSOLE (CONNECTS)
TO BOTTOM CONNECTOR OF
PROGRAMMER'S CONSOLE

F

J3
SERIAL
LINE UNIT

J4
PARALLEL I/O
RECEIVE

E

S1-1   S1-10

D

C

B

J1
PROGRAMMERS
CONSOLE

A

8071-1
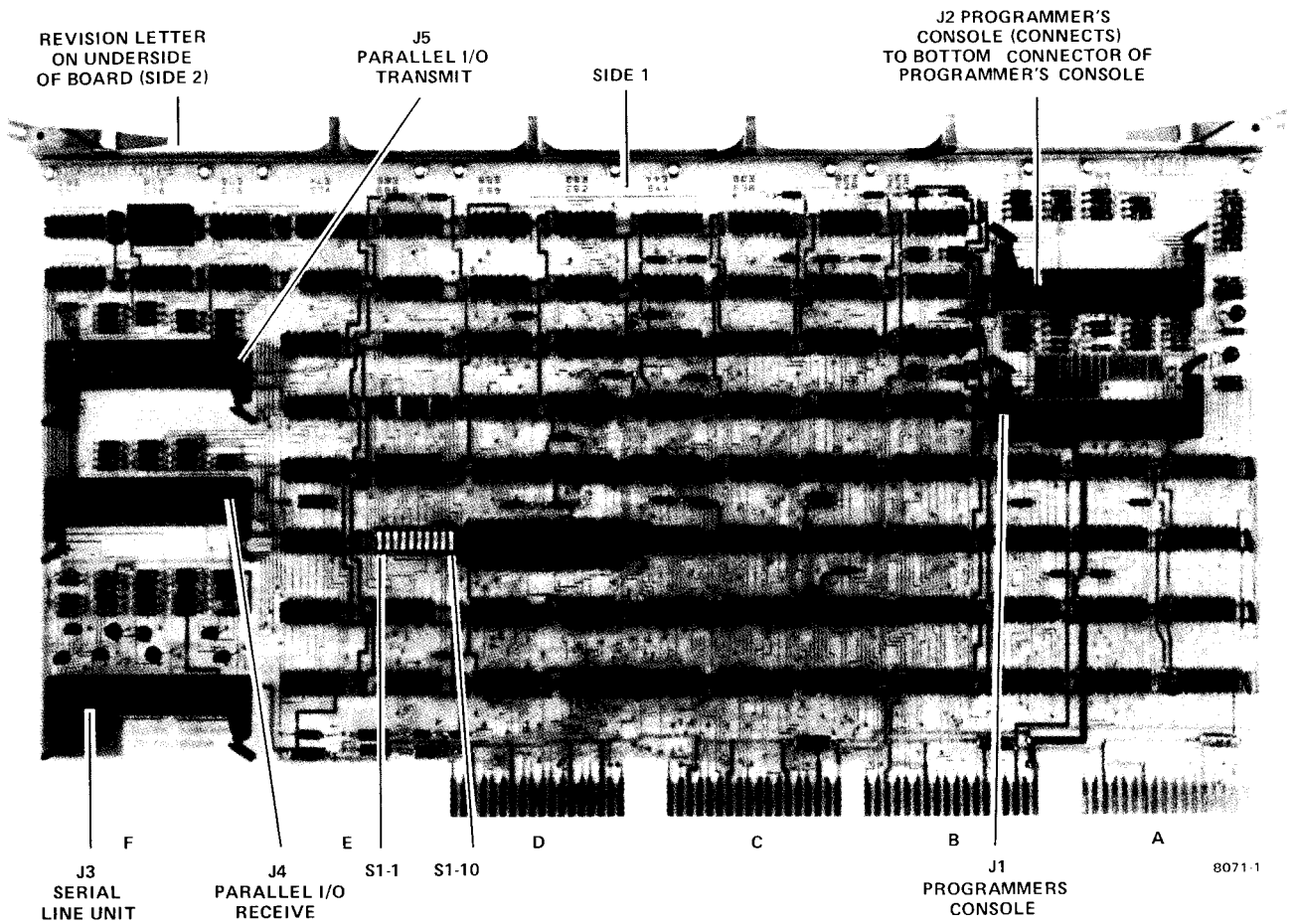
2.10 ENABLING PROPER TERMINAL CHARACTERISTICS

In order to perform proper terminal service, terminals and terminal interfaces must have proper settings. These settings control parity, stop bits, filtering and filler characters.

2.10.1 Parity

When a character is struck on a terminal connected to a PDP8, a bit pattern is sent to the computer via an interface. This bit pattern is sent in ASCII coding. However, ASCII standards specify only the seven-bit configuration for all characters. The high order eighth bit may be 0 or 1, depending on the settings of the terminal or the interface. Some of the software for the PDP8 ignores this high order bit. The remainder of the software assumes that this high order bit is always set ("eighth-bit always marking"). If this bit is not set, the character is ignored. *set 8th bit to "1"*

Most DEC terminals (e.g., ASR33, LA36, VT52) have an option to always set the high order bit to 1. The VT100 and many non-DEC terminals do not have this option. The latter terminals usually have options for odd or even parity. With odd parity, each character is transmitted with an odd number of bits. The high order bit is used to insure this fact. An "A" is represented as a 301, "B" is represented as 302, "C" is represented as 103, etc. With even parity, each character is transmitted with an even number of bits. "A" is represented as 101, "B" is represented as 102, "C" is represented as 303, etc. Both of these types of parity are unacceptable for software which assumes the high order bit is always on.

For terminals which do not have the capability to set the high order bit, you must set this bit on the interface board. The KL8E (M8650) and DKC8-AA (M8316) do not have this feature.

Therefore, you should not connect a VT100 or Hazeltine terminal
to one of these modules.

The KL8JA (M8655) and KL8A (M8319) boards contain this option.
To enable the option on a KL8JA board, insure that the

       NP jumper,
       NB1 jumper, and
       NB2 jumper

are removed from the board.   Refer to the KL8JA (M8655) prints
for the location of these jumpers.

2.10.2 Stop Bits and Filter

To enable the option on a KL8A board (M8319), insure that the

       NP jumper,
       NB1 jumper and
       NB 2 jumper

are removed from the unit for each line on which you wish to
generate eighth bit always mark.   Refer to the KL8-A User's
Manual for the location of these jumpers.

In addition to possible parity problems, you must consider
problems concerned with terminals running at 110 baud.
Interfaces operating at 110 baud must generate two stop bits
instead of one. A filter must also be enabled.

To enable a KL8-JA (M8655) to generate two stop bits and enable
the 110 baud filter, insure that the

       W4 jumper

is removed from the board and that the

W1 jumper

is installed on the board.   Refer to the KL8-JA Engineering
Drawings for the location of these jumpers.

To enable a KL8A (M8319) to generate two stop bits, insure that
the

2SB jumper

is removed from the board for each line on which you wish to run
a 110 baud terminal.   To enable the teletype filter for line 0,
set the S2-7 or S4-7 switch on for line H326 patch panel section
the KL8A is connected to.   To enable the teletype filter for line
1, set the S2-6 or S4-6 switch on.   To enable the teletype filter
for line 2, set the S2-5 or S4-5 switch on.   To enable the
teletype filter for line 3, set the S2-8 or S4-8 switch on.
Refer to the KL8-A User's Manual for the location of jumpers and
switches.

To enable the DKC8-AA (M8316) I/O option board to generate two
stop bits and enable the 110 baud filter, insure that

switch S1-7

is off and

switch S1-8

is on.   Refer to the PDP-8/A Operator's Handbook for the location
of these switches.

To enable a KL8E (M8650) to generate two stop bits and enable the
110 baud filter, insure that

jumper 2-3 in group J and .47 mfd capacitor

are installed and that

jumper 1-2 in group J

is removed.   Refer to the KL8-E Engineering Drawings for the location of these items.

2.10.3 Fill Characters

The final terminal problem to consider is fill characters. Most terminals print each character in the time it takes to generate the next character to print. However, some terminals require a time lag after special characters. A DECwriter equipped with the hardware form feed option requires a time lag after a form feed is output. To create this time lag, you must output a certain number of null characters to the terminal. The software can be enabled to generate any number of null filler characters after the output of a specified ASCII code (see 3.10). If you connect a device, requiring four filler characters after the generation of each line feed, to a KL8JA (M8655) module, these filler codes may be generated by the terminal interface. A VT05 operating at speeds above 300 baud requires four fillers after a line feed.

To enable this filler option on a KL8JA (M8655) module, insure that the FIL jumper is installed. Refer to the KL8-JA engineering drawings for the location of this jumper.

2.11 TESTING THE ETOS BOARD

The TSC8.SV program is intended for use as a diagnostic to test
the functioning of the ETOS board, properly referred to as
TSC8-75. In addition to merely testing for correct option, this
program provides console terminal output to aid in the diagnosis
of the specific fault in the board under test.

2.11.1 Installation Of The ETOS Board

The ETOS board is a quad module, which can be installed on any
omnibus PDP-8. The board takes up one slot and one ampere of
power at five volts. The device code of the TSC8-75 is 36. No
other device may use this code. Terminal interfaces are normally
the only boards which might conflict with the ETOS module.

On a PDP-8/E, /F or /M, the module can be installed in any open
slot between the processor boards and the shield board (M849).
The components should be at the front of the board when it is
plugged into the omnibus. Insure that no other board is touching
the TSC8-75.

On a PDP-8/A, the module can be installed in any open slot
between the memory boards and bottom of the box. The components
of the board should be face up when it is plugged into the
omnibus. Insure that no other board is touching the TSC8-75.

2.11.2 Hardware Requirements

The minimum hardware required to run the diagnostic is a PDP8/E/M
or /A with 4K of memory, a KM8E or KM8-AA with time sharing
enabled (see 2.7), the TSC8-75 board, and a console terminal.

To properly test the board, the diagnostic should be run with all
of the hardware in the final ETOS installation. This will test

the effect of any peculiar interactions between the specific pieces of hardware on the system.

The diagnostic <u>must</u> be run stand-alone. It cannot be run under ETOS.

2.11.3 Preliminary Diagnostic Programs

All of the Digital Equipment Corporation diagnostics should be run with the TSC8-75 in the system. Of these, perhaps the 8E ADDER TEST, EXTENDED MEMORY AND CHECKERBOARD, AND MEMORY EXTENSION AND TIME-SHARE tests are the most important. If any of these tests fail, it may be the fault of the TSC8-75 board. Remove it and try the test again; if it works this time, it is probably the TSC8-75 that is causing the problem.

In addition to all the DEC diagnostics, stand-alone OS/8 should be able to operate properly.

2.11.4 Operating Instructions

To initiate the diagnostic, boot a copy of the ETOS production pack and type <u>R TSC8<RET></u>. The system displays a program header and then prints the messge "ENTER SR AND PRESS RETURN:" and waits. At this point, a switch register setting must be selected. The format of the switch register is listed in Table 2-1.

Table 2-1
TSC8 SR Format

SR Bit   Function If Set

0-7   Use the corresponding memory fields for diagnostic tests.
          Field 0 is always used. Generally the user would set one
          of these bits for every memory field from 1 through 7
          that is physically present in the computer. For example,
          to indicate a 16K system, SR bits 0, 1, 2, and 3 would be
          set.

8   Halt the processor before displaying the "PASS COMPLETE"
          message. This option might be desired for unattended
          operation.

9   Test ESME option (see 3.5).

10   Instead of displaying characters on the console terminal,
          halt the processor with the ASCII code of the character
          in the AC. Pressing the CONT key on the front panel will
          continue with the diagnostic. This option will almost
          never be used. It is intended to allow limited operation
          of the diagnostic when no console terminal is available.

11   Retry all errors until they succeed. Normally when an
          error occurs, the diagnostic proceeds with the next trail
          in the sequence. This option retries the exact same
          operation under hopefully identical conditions until no
          error occurs. This would be useful to determine the
          consistency of the occurrence of the error, and in
          certain phases of board repair.

To initiate the diagnostic, enter your desired switch register as
a four digit octal number and press carriage return. A sample
execution of the diagnostic running on a 16K computer is listed
in Figure 2-24.

Figure 2-24
Sample Run of TSC8

.R TSC8
'TSC8-75' V2.003

ENTER SR AND PRESS RETURN: 7400

(diagnostic is now running)

If you wish to stop the diagnostic, halt or re-boot the system. The program destroys the core resident portion of OS/8, so that you cannot return to OS/8 by restarting at location 07600. The starting address of the diagnostic is 000200. If you have a front panel, you may restart the diagnostic at this location after it has been loaded.

2.11.5 Diagnostic Action

At this point, the diagnostic performs a series of tests successively in field 0 and then in each additional selected memory field. If any errors are detected, error messages are displayed. When this series is completed, the message "PASS nnnn COMPLETE. ERRORS DETECTED: nnnnnnnn" is displayed. The time required for each pass is a function of the number of errors that occur and the number of additional memory fields selected. If no errors occur, a typical pass might take about two minutes. Three or four passes are generally adequate to detect most board failures. Intermittent failures may require extended running times.

2.11.6 Keyboard Options

There are several console keyboard options available to control terminal output (see Table 2-2). They are turned ON by entering the appropriate control character. Only one option can be in effect at one time. Entering a new option cancels the old one, if any. Entering an unrecognized character also cancels the current option, but is otherwise ignored.

)

Table 2-2
TSC8 Console Keyboard Options

Character	Function When Entered
^S	Suspends terminal output until another character is entered. No output characters are lost.
^K	Deletes terminal output. Characters that would have been output are lost. This option is useful for suppressing unwanted output in cases where the ^O option is not adequate.
^O	Deletes terminal output until: (a) The end of the current pass; (b) Tests are started in a new memory field; (c) The type or kind of error message changes; (d) Another character is entered at the keyboard. This option is useful when a lot of similar error messages are being displayed and the user wishes to suppress them but not to suppress any other output.

Note that ^C is not supported, since the diagnostic destroys OS/8.

2.11.7 Error Messages

All error messages start at the beginning of the line with the test "?ERROR: ". Normally, the occurrence of any such message indicates a hardware failure in the system. This might not be the case if the operator was using a nonexistent memory field (see 2.11.4), or if an unusual device or static electricity causes a spurious or undefined interrupt.

The following messages do not usually indicate a hardware failure of the TSC8-75 board.

    "?ERROR: SPURIOUS INTERRUPT"

This could be caused by static electricity, or by a terminal

interface running open, or by some nonstandard device doing something unusual on the system. Unless it reoccurs, this message can probably be ignored.

"?ERROR: UNDEFINED INTERRUPT"

This is a somewhat more serious version of the previous error, since it probably will not go away. Although it may not indicate a hardware problem, it will prevent the diagnostic from running properly.

For the purposes of this diagnostic, any board that generates an interrupt and is not the KM8E Memory Extension and Timeshare Option (or KM8-A) or the TSC8-75 ETOS Time Share Control Module, is a special interface and must be inhibited from interrupting. Any such interrupt will cause one of the error messages in this section.

As distributed by QUODATA, the diagnostic program contains the code listed in Figure 2-25 to disable special interfaces.

Figure 2-25
Code to Disable Special Interfaces

01021	6035	KIE	/DISABLE INTERRUPT ON ALL
01022	6405	KIE&7007+400	/ STANDARD KL8E AND
01023	6425	KIE&7007+420	/ KL8J INTERFACES
01024	6445	KIE&7007+440	
01025	6465	KIE&7007+460	
01026	6305	KIE&7007+300	
01027	6325	KIE&7007+320	
01030	6662	PCLF	/CLEAR LINE PRINTER FLAG

The user may change this code at his discretion to correct any problems he may be having with special interfaces. The accumulator does not have to be left clear. Locations 01150 through 01173 may be used as additional patch space. Interfaces which would be likely to be special interfaces include

(a) Any laboratory or customized interface.
(b) Any interface with a device code that conflicts with the standard codes listed above.
(c) Any KL8A. This really falls into category (b). The instruction to disable a KL8A with device codes 40/41 is 6400. The instruction to disable a KL8A with device code 42/43 is 6420.

"?ERROR: TIME SHARE OPTION NOT ENABLED
DIAGNOSTIC CANNOT CONTINUE"

The processor halts at this point unless the "retry error" switch register option is set (see 2.11.4). Pressing the CONT key on the computer front panel will retry the test. If the time share option has been enabled, something is seriously wrong with the KM8-E, the KM8-A, or the interrupt logic. The user should determine what is the cause of the error, or call Digital's Field Service.

Any error message other than the ones mentioned previously indicate a hardware problem with the TSC8-75 module (unless nonexistent memory has been specified. Many of these messages occupy more than one line of text. Run the diagnostic until several feet of hard copy output has accumulated, preferably through at least three or four passes. It is better if you can obtain several different types of error messages. If the diagnostic seems to be outputting the same message repeatedly, you might try using the keyboard options "^O" and "^K" (see 2.11.6). Record the value of the switch register on the output and send it to QUODATA along with the defective TSC8-75 module.

A list of the initial text of some of the hard error messages is given in Table 2-3. This text, along with the messages listed in this section, identifies the type or kind of error referred to in the "^O" option.

Table 2-3
TSC8 Hard Error Messages

Error Message

?ERROR: ECTF SETS FLAG
?ERROR: ECTF DOESN'T CLEAR FLAG
?ERROR: IMPOSSIBLE SKIP
?ERROR: nnnn REGISTER FAILURE(S)
?ERROR: MEMORY FAILURE ON JMS
?ERROR: BAD AC
?ERROR: BAD AC

CHAPTER 3
CONFIGURING THE SYSTEM

3.1 INTRODUCTION

QUODATA delivers the same version of ETOS to all sites. The
CONFIG program which runs only under stand-alone OS/8 is used to
tailor an ETOS system to a particular installation. The software
changes in the QUODATA version of ETOS necessary to configure to
your installation must be done before ETOS is initiated.

If necessary, the CONFIG program may be run again at any later
date in order to implement a change in the hardware system.

3.2 CONFIG OPERATING INSTRUCTIONS

To start the CONFIG program, enter R CONFIG<RET> after stand-
alone OS/8 has been booted. CONFIG responds with a program
header and an asterisk (*).

There are three ways in which ETOS can be altered to interface
properly with the hardware of your unique system.

First, you may specify a binary patch that will enable the
hardware of your system. CONFIG prints out the changes effected
by the patch. The binary patch is called for by typing its
specification in reply to the "*" of CONFIG. A list of these
binary patches is given in Table 3-1. For example, to enable the
DK8EA line frequency clock enter

    *DK8EA

Patches exists for configuring the common clocks, line printers,
version of ETOS boards and for start-up procedures. These will
be discussed in later sections of this chapter.

Table 3-1
CONFIG Binary Patches

Specifications	Meaning	Manual Section
AUTO	Enable automatic command after option is entered.	3.7
DKC8AA	Enable DKC8AA real time clock.	3.3
DK8EA	Enable DK8EA line frequency clock.	3.3
DK8EC	Enable DK8EC crystal clock.	3.3
DK8EP	Enable DK8EP programmable clock.	3.3
ESME	Enable ESME feature on the TSC8-75 module.	3.5
KL8J	Enable line printer connected to a KL8E or KL8JA interface.	3.4
LA180D	Enable LA180 line printer connected to a DKC8-AA interface.	3.4
LP05	Enable LP05 type line printer.	3.4
LS8E	Enable LS8E type line printer.	3.4
NOAUTO	Disable automatic command facility (opposite of AUTO).	3.7
NOOPTN	Disable "OPTION?" question when starting ETOS (opposite of OPTN).	3.6
NOESME	Disable ESME feature for TSC8-75 modules (opposite of ESME).	3.5
OPTN	Enable "OPTION?" question when starting ETOS.	3.6

The second type of change made to ETOS involves the alteration of existing ETOS tables. These tables are listed in Table 3-2. These subroutines of CONFIG are used to ignore special characters, clear unimplemented device IOTs, set memory

configuration, set terminal handling, change initialization characters, and fine tune the scheduler. The format of these table changes is

    *table,offset=new value

where "table" is a table specification from Table 3-2, "offset" is a two digit octal code appropriate for the specific change and "new value: is a four digit octal code appropriate for the change. The codes are discussed in the appropriate sections of this chapter as indicated by Table 3-2.

All table specifications which modify ETOS are "traced"; that is, the table name, offset, the old value and the new value are printed. The table options documented in Sections 3.8-3.12 must be specified for ETOS to run properly. The remaining table options documented in 3.13-3.17 are used to automate the start-up procedure and tune ETOS to a particular job mix. They are not critical to the initial implementation of the system.

The third type of change is an alteration of any location in a component of ETOS.SV (excluding core control blocks). You should not use this form, unless the specifications come from QUODATA via the Software Maintenance Reports (SMRs) (see 8.12). Since you are modifying the monitor, a knowledge of the ETOS source is required to devise these specifications. All valid monitor components are listed in Table 3-3.

Table 3-2
CONFIG Tables

Specifications	Meaning	Manual Section	Length (Words)
BUFMAX	Set the size of the terminal input/output buffer.	3.14	1
BUFMIN	Set output buffer control specification.	3.15	1
FILLER	Generate null filler codes for specified terminals.	3.10	16
IGNCHR	Disable echoing of special character for specified terminals.	3.11	16
INITLN	Specify command line, which is automatically executed if AUTO option is enabled.	3.16	24
IOTTAB	Specify device codes of all terminals.	3.9	16
LSLICE	Tune ETOS scheduler to a particular job mix.	3.13	8
MEMTAB	Specify available memory fields.	3.8	8
UNDINT	Specify undefined devices generating interrupts.	3.12	8
XOFFCT	Set input buffer control specification.	3.16	1

Table 3-3
ETOS Monitor Components

Component	Meaning
RMON	Resident Monitor
KMON	Keyboard Monitor
DMON	Disk Monitor
SYSG	Sysgen

To change a word in a component of the monitor, enter

 *COMPONENT,address=new value

where "component" is a specification from Table 3-3, "address" is
the octal location in ETOS which you wish to modify. If the
address is less than five octal digits, the modification is
performed in field 0. If the address is five digits, the first
digit is taken as the field to modify. "new value" represents
the new contents of the specified location.

After entering a command in CONFIG, the system will print any
changes and continue responding with an "*" after each
specification line. You can exit the program by typing EXIT<RET>
or ^C. Any changes specified have been made to ETOS.

All specifications that modify ETOS are "traced", that is, the
component, the field and four digit address, the old value and
the new value are printed.

If you desire to list a table entry, enter the table name and
offset followed by "/L". Figure 3-1 illustrates this procedure.

Figure 3-1
Sample Use of /L to List a Table Entry

 *IOTTAB,00/L
 IOTTAB(00000) 6045
 *

To list an entire table, enter the table name followed by "/L",
with no offset as shown in Figure 3-2.

Figure 3-2
Sample Use of /L to List an Entire Table

```
*MEMTAB/L
MEMTAB(00000)  0001
MEMTAB(00001)  0001
MEMTAB(00002)  0000
MEMTAB(00003)  0000
MEMTAB(00004)  0000
MEMTAB(00005)  0000
MEMTAB(00006)  0000
MEMTAB(00007)  0000
*
```

When entering table entries or monitor component changes, you may
use an alternate format of the CONFIG commands, which does not
modify a location but instead opens it for possible modification.
This form uses no equal sign or any specification which normally
is input to the right of the equal sign. After the command is
specified, the current contents of the location is printed and
CONFIG waits. If you wish to change the contents, enter the new
contents followed by either a carriage return when this is the
last entry in the table which you wish to modify or a line feed
if you also wish to modify the next entry. Terminate your input
with a minus sign (-) to display the preceding location or a
slash (/) to display the current location. Enter carriage return
or line feed if you do not wish to modify a particular location.
A sample configuration of a memory table using the CONFIG display
syntax follows in Figure 3-3.

Figure 3-3
Sample Use of CONFIG Display Syntax

```
.R CONFIG<RET>
'CONFIG' V3.015
*MEMTAB,2<RET>
MEMTAB (00002)/0000:  <LF>
MEMTAB (00003)/0000:  <LF>
MEMTAB (00004)/0000:  0001<LF>
MEMTAB (00005)/0000:  0001<LF>
MEMTAB (00006)/0000:  0001<LF>
MEMTAB (00007)/0000:  0001<RET>
*
```

Note: If CONFIG is being run under BATCH, the letter "C" should
         be used in place of <RET> and the plus sign should be
         used in place of <LF>.

Instead of entering commands from the terminal, you may create
batch streams to enter specifications. CONFIG may be run under
BATCH, since it uses the command decoder. The dialogue of Figure
3-4 can be used to create and submit a batch stream to configure
the type of clock and the amount of memory contained in the
system.

<div align="center">

Figure 3-4
Sample Use of CONFIG Under BATCH
</div>

```
.CREATE CONSPC.BI
#A
$JOB TO CONFIGURE CLOCK AND MEMORY
.R CONFIG
*DK8EA
*MEMTAB,02=0
*MEMTAB,03=0
*MEMTAB,04=0
*MEMTAB,05=0
*MEMTAB,06=0
*MEMTAB,07=0
*EXIT
$END
<FF>

#E
.SUBMIT CONSPC.BI
```

Note: <FF> stands for the form feed <CTRL/L> character.

When the system is delivered, the initial parameters may be set
to any value. You must configure all options to insure that ETOS
will run properly.

3.3   CLOCK OPTIONS

There are four types of clocks which Digital Equipment
Corporation offers for the PDP-8. To configure ETOS for one type

of clock, enter the respective designation for the clock in response to the "*". The types of clocks are as listed in Table 3-4.

Table 3-4
Clock Options Under CONFIG

Option	Installed
DK8EA	DK8-EA line frequency clock
DK8EC	DK8-EC crystal Clock
DK8EP	DK8-EP programmable clock
DKC8AA	DKC8-AA PDP-8A real time clock

A listing of clock service is provided in the file CLOCK.LS on the distribution pack in account [0,10]. If you do not have one of the above types of clocks, you will have to implement your clock with the documentation provided in the listing. To list the clock service on a terminal, log in to account [0,10] on a copy of the distribution pack and enter TYPE CLOCK.LS<RET>.

If you bring ETOS up and press carriage return on any terminal, the system should respond with "?LOGIN PLEASE". If it takes seven or eight carriage returns to trigger the LOGIN message, you have probably enabled the wrong clock option or there is a problem with the physical clock. If you run ETOS for a period of time and the system is not keeping the right time, you have enabled the wrong clock or your clock is running at a non-standard rate.

ETOS assumes that the DK8-EA clock runs at 120 Hz. However, some DK8-EA clocks run at 60Hz. If you have a DK8-EA clock running at 60Hz, use the procedure of Figure 3-5 under stand-alone OS/8 to eliminate the problem.

Figure 3-5
Changing CONFIG to Support 60 Hz DK8-EA

```
.R FUTIL
FILE CONFIG
CONFIG.SV  ssss-eeee  0012  (0010)  b.111 da-mon-yr
SET MODE SAVE
20222/  7764   7772<LF>
000.20223\  7764  7772<RET>
WRITE
EXIT
.
```

After this procedure is followed, you must enable the DK8EA option with the revised CONFIG.

3.4 LINE PRINTER OPTIONS

To configure ETOS for one of the four types of PDP-8 line printers, enter the respective designation for the line printer from Table 3-5.

Table 3-5
Line Printer Options Under CONFIG

Option	Installed
LS8E	Parallel line printer connected to an LS8E (M8342) interface. A parallel LA180 connected to an M8365 module (LA8-PA) is program compatible with the LS8E.
LP05	LP05 line printer connected to LE8 interface.
KL8J	Serial line printer connected to a KL8E (M8650) or KL8JA (M8655) interface.
LA180D	Parallel LA180 line printer connected to the DKC8-AA I/O option board #1 (M8316).

A listing of line printer service is provided in the file LPSERV.LS on the distribution pack in account [0,10]. If you do not have one of the above types of line printers you will have to implement your line printer with the documentation provided in

the listing. To list the line printer service on a terminal, LOG
IN to account [0,10] on a copy of the distribution pack and enter
TYPE LPSERV.LS<RET>.

3.5 ESME FEATURE OF THE ETOS [TSC8-75] MODULE

Later revisions of the TSC8-75 module contain a feature called
ESME which increases throughput of some processes by five to ten
percent. All instructions which refer to memory fields must be
mapped by the ETOS software to refer to an available physical
memory field. This mapping introduces an overhead factor to all
memory field instructions. The ESME feature checks to see if the
Change Data Field (CDF) instruction being executed is referring
to the same field as the previous change instruction. If the
fields are the same, the instruction is ignored and no software
mapping occurs. This is where the increased efficiency is
achieved.

All ETOS boards (TSC8-75) which have a serial number greater than
699 contain the ESME feature. Some boards with lower serial
numbers contain this feature, because you have upgraded your
module. If you are unsure as to whether or not your board
contains this feature, the TSC8 diagnostic (see 2.11) may be used
to determine this. If you disable the option, no harm will occur
even if the board contains the ESME feature. You will simply
lose a small percentage of efficiency.

The two options concerning the ESME feature are listed in Table
3-6.

Table 3-6
TSC8-75 Options Under CONFIG

Option	Meaning
ESME	Enable ESME feature
NOESME	Disable ESME feature

The patch options documented in 3.2-3.5 must be specified for ETOS to run properly. The remaining patch options, documented in 3.6 and 3.7, are used to automate the start-up procedure and are not critical to the initial implementation of the system.

3.6   OPTION QUESTION IN START-UP PROCEDURE

When you initiate ETOS from stand-alone OS/8 (see 4.5), the system prints "OPTION?". There are options for returning to stand-alone OS/8, recreating the ETOS file structure and restoring a new copy of the virtual OS/8 system head. These are special purpose options normally used when you're installing the system or recreating a destroyed part of the system. The option used on a daily basis is option "T", which starts time sharing.

If you disable the OPTION? question, the system automatically comes up in time sharing mode after you enter R ETOS<RET> under stand-alone OS/8. The two specifications concerning the option question are listed in Table 3-7.

Table 3-7
Start-up Options Under CONFIG

Option	Meaning
OPTN	Enable OPTION question in start-up procedure.
NOOPTN	Disable OPTION question. Starts timesharing mode.

3.7 AUTOMATIC START-UP

When you initiate ETOS from stand-alone OS/8 (see 4.5), the system prints "OPTION?". When you enter T<RET>, timesharing is started. At this point, you are ready to LOGIN to an account. The system has the capability to automatically execute a command after the OPTION question is answered. This command is forced into the input buffer of the console terminal and prints out while it is being forced. The command is normally a LOGIN command which allows you to bring timesharing up with the console terminal logged in. The command which can be automatically executed is set up with the INITLN option of CONFIG (see 3.17). The options in this section are used to disable or enable the automatic command execution. The two options which control auto-start are listed in Table 3-8.

Table 3-8
Automatic Start-up Options Under CONFIG

Option	Meaning
AUTO	Enable the auto-start feature. Execute the command specified in the INITLN table after the "OPTION?" question is answered.
NOAUTO	Disable the auto-start feature.

3.8 MEMORY CONFIGURATION

The memory allocation table contains one entry for each 4K word memory bank which might be contained in the system. There are eight entries, which allows up to 32K words.

The general form of the command is

    *MEMTAB,offset=indicator

The entries are in order, so that the offset numbers in the

command correspond with the physical field numbers. If the "indicator" value is set to zero, the field is available for use under ETOS. If the entry is set to one, the field is not present in your configuration. One exception is that field 0 and 1 are always set to one, so that they are reserved for the resident monitor of ETOS (RMON). They cannot be changed. If you try to change the entry for field 0 or 1, the message "OFFSET OUT OF RANGE" is printed. This implies that at least two other entries must be zero to run ETOS, since four fields (16K) are required.

To set up a machine for 20K, the dialogue of Figure 3-6 is used.

<div align="center">

Figure 3-6
Sample Use of the MEMTAB Option Under CONFIG

</div>

```
*MEMTAB,02=0
MEMTAB(00002)   0000   0000
*MEMTAB,03=0
MEMTAB(00003)   0000   0000
*MEMTAB,04=0
MEMTAB(00004)   0000   0000
*MEMTAB,05=1
MEMTAB(00005)   0000   0001
*MEMTAB,06=1
MEMTAB(00006)   0000   0001
*MEMTAB,07=1
MEMTAB(00007)   0000   0001
*
```

3.9 TERMINAL CONFIGURATION

The terminal table contains the device code of each terminal on the system.

The general form of this command is

    *IOTTAB,offset=terminal IOT

The offset value for the table corresponds to the console numbers displayed when LOGging INto the system.

There are sixteen entries in this table, which allows sixteen jobs on the system. If a physical terminal does not exist for a given offset, the entry should be set to zero. This will allow you to detach jobs, since the detach processor uses consoles with an entry of zero to associate with the detached job. The device code has two forms.

The terminal IOT value is determined in the following way. If the terminal is attached to a KL8E or KL8-JA, the form of the terminal IOT is 6XX5, where "XX" represents the teleprinter device code. If the terminal is attached to a KL8A, the form of the terminal IOT is 0XXY, where "XX" represents the device code of the KL8A and "Y" represents the line number (from 0 to 3).

To set up a ten terminal system with a console interface (03/04), a KL8-JA with a device code of 30/31, a KL8A with a device code of 40/41, and a KL8A with a device code of 42/43, enter the commands contained in Figure 3-7.

Figure 3-7
Sample Use of the IOTTAB Option Under CONFIG

```
*IOTTAB,00=6045
IOTTAB(00000)   6045   6045
*IOTTAB,01=6315
IOTTAB(00001)   0400   6315
*IOTTAB,02=0400
IOTTAB(00002)   0401   0400
*IOTTAB,03=0401
IOTTAB(00003)   0402   0401
*IOTTAB,04=0402
IOTTAB(00004)   0403   0402
*IOTTAB,05=0403
IOTTAB(00005)   6475   0403
*IOTTAB,06=0420
IOTTAB(00006)   0000   0420
*IOTTAB,07=0421
IOTTAB(00007)   0000   0421
*IOTTAB,10=0422
IOTTAB(00010)   0000   0422
*IOTTAB,11=0423
IOTTAB(00011)   0000   0423
*IOTTAB,12=0
IOTTAB(00012)   0000   0000
*IOTTAB,13=0
IOTTAB(00013)   0000   0000
*IOTTAB,14=0
IOTTAB(00014)   0000   0000
*IOTTAB,15=0
IOTTAB(00015)   0000   0000
*IOTTAB,16=0
IOTTAB(00016)   0000   0000
*IOTTAB,17=0
IOTTAB(00017)   0000   0000
```

Xmit device code only

The offset value specified in this command is also used when initiating the SPOOLR to access a terminal as a spooling output device. When the SPOOLR asks for the physical device, type TTXX where XX corresponds to the offset for the output terminal. See 6.4.13 in the Manager's Guide for additional details.

This offset value is also the console number used in the FORCE and ECROF commands (see 6.2.4) and the TALK command (see 3.9, System User's Guide).

If you are crashing frequently, a bad terminal or terminal interface is often the source of these crashes. If you suspect that a terminal is causing problems, remove the entry for it in the IOT TABle. For example, if terminal 02 is suspect, enter under CONFIG

    *IOTTAB,02=0

After removing the IOT entry, set one of the undefined interrupt entries to 6XX5, where XX is the keyboard device code of the interface. If the system does not crash after this process, have the terminal or its interface serviced.

3.10   FILLER CHARACTERISTICS

Some terminals require null filler characters after generation of a special character such as carriage return, line feed, vertical tab or form feed. The filler table allows you to generate up to fifteen null filler characters after output of a particular ASCII character. One filler specification is available for each terminal on the system. Once again, the offset value indicates the terminal number. There are sixteen entries in the filler table and the terminals are in the same order as they are in the terminal IOT TABle (see 3.9). Set the filler entry to zero to generate no fillers for a particular terminal. The general form of the command is

    *FILLER,offset=filler specification

The filler specification is a twelve-bit word. The first eight bits make up the ASCII code after which you wish to generate filler characters. The value of its parity bit is irrelevant. The last four bits are the number of nulls to generate after output of the respective ASCII character. To generate five nulls each time a form feed (ASCII code 214) is output on terminals 00,

04 and 10, enter the commands of Figure 3-8.

Figure 3-8
Sample Use of the FILLER Option Under CONFIG

```
*FILLER,00=4305
FILLER(00000)  0000  4305
*FILLER,04=4305
FILLER(00004)  0000  4305
*FILLER,10=4305
FILLER(00010)  0000  4305
*
```

3.11 IGNORING SPECIAL CHARACTERS

Some terminals perform functions using the terminal hardware that
is normally performed with software. An example of this type of
function is a hardware tab. Normally, a tab (ASCII 211) is
interpreted by the software as the number of spaces required to
position the terminal at the next column which is a multiple of
eight. The hardware tab positions the terminal at the next stop
and the software spaces to the next tab stop. This results in
double echoing of tabs on terminals that contain the hardware tab
feature. ETOS has the capability of not echoing certain
characters. If the user sets a tab as one of the characters not
to echo, then tabs will be processed correctly on a terminal with
hardware tabs. Two characters can be selected for this purpose.
The general form of the command is

*IGNCHR,offset=ASCII value of character to ignore

ETOS can be enabled to ignore up to two characters. The "offset"
is either 00 or 01. The parity bit ("200" bit) of the character
must be set. If you wish to change one of the entries in the
table to not ignore any character, set the entry to 4000. To set
ETOS to ignore tabs, but no other character, the sequence of
Figure 3-9 is used.

Figure 3-9
Sample Use of the INGCHR Option Under CONFIG

```
*IGNCHR,00=211
IGNCHR(00000)   0000   7567
*IGNCHR,01=4000
IGNCHR(00001)   0000   4000
*
```

When you use this option, CONFIG automatically sets the table entry to the negative of the ASCII value. ETOS requires it in this form. You do not need to be concerned about this negation.

Set the values in this two entry table even if the tables does not ignore any characters.

Note: If this feature is used, the selected characters are ignored by all terminals in the system.

3.12 UNDEFINED INTERRUPTS

You may have devices on your system which ETOS does not use, but which you use under stand-alone OS/8. An example of such a device is the Floating Point Processor (FPP). A problem occurs in that even though ETOS does not use these devices, they may generate interrupts. Therefore, the user should insert into the undefined interrupt table, the code which clears the interrupt for that device. To determine the code which clears the interrupt for your device, reference the technical manual for that hardware. Codes for most Digital Equipment Corporation devices are contained in the Small Computer Handbook or the PDP8/A Miniprocessor Handbook. If this is not done, the unimplemented device will crash ETOS. The general form is

```
*UNDINT,offset=clear interrupt IOT
```

The "offset" may be from 00 to 07, since the user may clear up to eight unimplemented devices. For example, to clear the interupt

on the XY8-E Incremental Plotter and four DP8E Synchronous Modem Interfaces, enter the commands of Figure 3-10.

Figure 3-10
Sample Use of the UNDINT Option Under CONFIG

```
*UNDINT,00=6500
UNDINT(00000)  7000  6500
*UNDINT,01=6401
UNDINT(00001)  7000  6401
*UNDINT,02=6421
UNDINT(00002)  7000  6421
*UNDINT,03=6441
UNDINT(00003)  7000  6441
*UNDINT,04=6461
UNDINT(00004)  7000  6461
*
```

If you wish to change an entry, which formerly contained a clear IOT, and not disable any device, change the entry to 7000.

3.13 TUNING THE ETOS SCHEDULER

Each job on the system is run for a specified period of time before it is swapped out (if necessary). Another job is run for its prescribed period of time. The last digit of the privilege word (see 6.2.1) controls how long this period or quantum is. When a user logs in, the system sets his privilege word to 0040. The last "0" controls the quantum. This digit is an index into an eight word table controlled by the LSLICE option. The general form of the command is

*LSLICE,offset=-number of system ticks

The "offset" is in the range 0 to 7, corresponding to the last digit of the privilege word. The table entry is set equal to the negative number of system ticks for which the job runs before another job is scheduled. Each system tick is a tenth of a second (millisecond). Therefore, to cause all jobs with a

privilege word equal to XXX4, to run for five and a half seconds (55_{10} or 67_8 system ticks) before another job is scheduled, enter the commands of Figure 3-11.

Figure 3-11
Sample Use of the LSLICE Option Under CONFIG

```
*LSLICE,04=7014
 LSLICE(00004) 7670 7014
*
```


The system is delivered with an optimum scheduler table for the average job mix.   Table 3-9 contains a list of the initial entries.

Table 3-9
Initial Entries in the LSLICE Table

Entry	Contents	Quantum
0	7730	50 (base 8) or 40 (base 10) tenths
1	7720	60 (base 8) or 48 (base 10) tenths
2	7710	70 (base 8) or 56 (base 10) tenths
3	7700	100 (base 8) or 64 (base 10) tenths
4	7670	110 (base 8) or 72 (base 10) tenths
5	7660	120 (base 8) or 80 (base 10) tenths
6	7650	130 (base 8) or 88 (base 10) tenths
7	7640	140 (base 8) or 96 (base 10) tenths

If your job mix varies significantly from an average system which has an equal number of computation, terminal input/output, line printer output and disk input/output, use this option to change the initial values. A job which is performing strictly terminal input/output at less than 5800 baud should have a small quantum, since the job will send or receive the data in an in-core buffer and dismiss the job to perform the actual I/O. A job which is performing strictly disk input/output should have a larger quantum, since a greater number of disk transfers performed in one quantum will result in less revolutions of the disk. A job which is not time-critical should have a small quantum which will

reduce overhead for other jobs.

3.14 TERMINAL BUFFERS

When ETOS is running, the resident monitor is always contained in field 0 and 1. Most of field 1 is used for buffer space. This space which contains the line printer buffer, terminal buffers and ETOS file buffers is known as free core. If you have more than seven or eight jobs active, where each job is using a number of buffers, you may run out of free core. The system will stop processing and you will have to restart ETOS. To eliminate this problem, the BUFMAX option has been provided. The general form of this command is

   *BUFMAX=buffer size

BUFMAX is a one word table. "Buffer size" is the octal size of the input/output terminal buffer. This option affects all terminals on the system. When the system is distributed, the size of the terminal buffer is 372 (base 8) or 250 (base 10) characters.

To set the terminal buffer size for a system using eleven jobs with a significant amount of input/output, use the dialogue of Figure 3-12.

Figure 3-12
Sample Use of the BUFMAX Option Under CONFIG

```
*BUFMAX=175
BUFMAX(00000) 0372 0175
*
```

Since each installation is unique, you have to experiment with this option (if you are crashing) to determine the proper buffer size for your job mix. If you change the value of BUFMAX, you

may want to change the value of BUFMIN (see 3.15) and XOFFCT (see 3.16).

3.15 TERMINAL OUTPUT

When a job prints output to a terminal, the output first is queued to an in-core buffer in field 1. The job is then dismissed from running (put to sleep). The ETOS resident monitor performs the output of the characters to the physical terminal. This process increases efficiency dramatically, because it reduces swapping. Since the resident monitor is always in memory, you perform no swapping while the characters are being printed. When the buffer empties to a certain level, the system restores the job to the run queue (wakes it up). While the remainder of the buffer is being printed, the job can continue processing or output more characters to the buffer. The BUFMIN option is used to set the level at which the system restores the job to the run queue. The general form of the command is

*BUFMIN=buffer level

BUFMIN is a one word table. "Buffer level" is the octal number of characters left in the buffer when the job is to be awakened. This option affects all terminals on the system which operate at a speed of less than 580 characters per second (5800 baud). Output is never buffered for terminals running at speeds greater than 5800 baud. Therefore, this option has no effect on these terminals. Another implication of this fact is that reducing terminals in speed from 9600 baud to 1200 or 1400 baud increases total system throughput.

When the system is distributed, the wake-up level is set at 113 (base 8) or 75 (base 10) characters. A large value of BUFMIN ensures that you won't see a pause on your output terminal, but this increases system overhead. A small value of BUFMIN may

result in terminal pauses, but it may increase system throughput. For example, if you want to ensure that your terminals output almost continuously, the dialogue of Figure 3-13 might be used.

Figure 3-13
Sample Use of the BUFMIN Option Under CONFIG

```
*BUFMIN=0300
BUFMIN(00000)  0113 0300
*
```

BUFMIN should be used in conjunction with BUFMAX (see 3.14). If the value of BUFMIN equals the value of BUFMAX, no terminal buffering will be performed by the system. Naturally, BUFMIN must always be less than BUFMAX.

3.16 TERMINAL INPUT

When a program accepts input from a terminal and the KSTAT word (see System User's Guide, 3.7.6) has bit 0 set, the ETOS resident monitor (RMON) performs the input. ETOS echoes the characters and outputs them to a terminal buffer in field 1. When a BREAK character is entered (see System User's Guide, 3.7.5), the program receives input from the terminal buffer. This buffering operation increases system efficiency dramatically, because it reduces swapping. Since the resident monitor is always in memory, you perform no swapping while the characters are being input. This method also allows you to enter commands, while the program is processing, which will be accepted when input is requested. This feature is knowns as type-ahead. There is a limit to the number of characters which may be buffered. When the number of characters buffered comes within a certain number of characters from the end of the terminal buffer (see 3.14), the system sends an XOFF (^S) to the terminal. The terminal will not accept any additional characters at this point. Any additional characters entered will not be echoed and the bell on the

terminal will ring. This XOFF process is necessary to insure that characters are not echoed which will be ignored. The XOFFCT option is used to set the number of characters from the end of the terminal buffer, which will trigger the XOFF. The general form of the command is

      *XOFFCT=buffer space

XOFFCT is a one word table. "buffer space" is the amount of space left in the buffer when the XOFF is sent. This option affects all terminals on the system.

When the system is distributed, the buffer space is set to 10 (base 8) or 8 (base 10) characters. A large value of XOFFCT insures that no characters will be echoed which will be lost due to the time lag between sending the XOFF and the terminal stopping input, but system overhead is increased. A small value of XOFFCT decreases system overhead, but characters may be echoed and not accepted. If bit 0 of KSTAT is set to 0 or XOFFCT is set to equal to BUFMAX (see 3.14), no input buffering is performed.

3.17 AUTOMATIC COMMAND LINE

When ETOS is initiated and the "OPTION?" question is answered, time sharing is up and running. Normally, the system manager must LOGIN to an account and perform daily start-up operations (e.g., initializing the time and date, mounting public packs, etc.) However, if you set the AUTO option (see 3.7), the system will execute one command line from the console terminal after the "OPTION?" question is answered. This command line is normally a LOGIN or HELLO command which LOGS you IN to the OPERATOR's account ([0,3]). Since you are LOGged OUT when the command is executed, the only commands which may be executed are HELLO, LOGIN, SYSTAT, VERSION and ATTACH. The INITLN option is used to specify the initial command. The general form of the option is

*INITLN,offset=ASCII code

INITLN is a 30 (base 8) or 24 (base 10) word table. Each entry contains an ASCII code which represents one character in the command. The command is formed by appending each character starting with offset 00. The command must be terminated by a 215 (<RET>). After the last entry in the command, you must set the next entry to 0000. The initial values in the INITLN table are listed in Table 3-10.

Table 3-10
Initial Value of the INITLN Table

Offset	Entry	Character Value
00	0310	H
01	0273	;
02	0260	0
03	0254	,
04	0263	3
05	0240	space
06	0317	O
07	0320	P
10	0305	E
11	0322	R
12	0301	A
13	0324	T
14	0317	O
15	0322	R
16	0215	<RET>
17	0000	terminator
20	0000	
21	0000	
22	0000	
23	0000	
24	0000	
25	0000	
26	0000	
27	0000	

The initial command specification is H;0,3 OPERATOR<RET>. This command LOGs you INto the operator's account. If you had changed the password of account [0,3] to SYSTEM, the dialogue of Figure

3-14 would be used to specify the INITLN.

Figure 3-14
Sample Use of the INITLN Option Under CONFIG

```
*INITLN,06=323
INITLN(00006)   0317   0323
*INITLN,07=331
INITLN(00007)   0320   0331
*INITLN,10=323
INITLN(00010)   0305   0323
*INITLN,11=324
INITLN(00011)   0322   0324
*INITLN,12=305
INITLN(00012)   0301   0305
*INITLN,13=315
INITLN(00013)   0324   0315
*INITLN,14=215
INITLN(00014)   0317   0215
*INITLN,15=0
INITLN(00015)   0322   0000
*
```

After you use the INITLN option to automatically LOG you IN, you can use the SET SYS INIT option of OS/8 to automatically execute a command or a batch file. See 6.4.12 for details on using SET SYS INIT.

CHAPTER 4

START-UP OPTIONS

4.1   INTRODUCTION

After configuring ETOS to your installation with the methods
presented in Chapter 3, you are ready to run ETOS.   At this
point, you have a one-user ETOS system.   Chapter 5 documents how
to allow more than one user to access the system.   This chapter
tells you how to specify the amount of swapping space you need,
how to start ETOS, how to initialize time and date and how to
stop ETOS.   In doing these adjustments, you run the ETOS system
from the console terminal only.

Single user ETOS is started from stand-alone OS/8.   To obtain
ETOS, type the instructions of Figure 4-1 in response to the OS/8
monitor ".".

Figure 4-1
Starting ETOS

```
.R ETOS
ETOS V5B
OPTION?
```

ETOS monitor responds as indicated requesting "options".   These
options are discussed in Section 4.2 to 4.5.

4.2   SYSGENING A PACK

The resident monitor of ETOS reserves 8K words (fields 0 and 1)
for its processing.   The rest of the physical memory in the
machine is available for user programs.   On a 32K word machine,
24K words is available for use.   If you run three 8K jobs, ETOS
simply brings them into memory and runs each job in turn.

If you run a fourth 8K job, there is no room to keep all jobs in

memory. ETOS must save one of the jobs on the disk. When the job which is residing on disk is ready to run, it must be exchanged with one of the jobs in memory. The job in core memory is saved on the disk. This process is known as swapping. Almost all of the ETOS overhead originates from the swapping and the mapping of the core fields.

ETOS must have a file on disk where it can swap users in and out of memory. The SYSGEN option is used to tell how much space you need for your system swap file. To use this option, boot stand-alone OS/8 on the ETOS pack and follow the dialogue of Figure 4-2

Figure 4-2
ETOS Sysgen Option

.R ETOS
ETOS V5B
OPTION? S   [for SYSGEN]
ARE YOU SURE? Y
NUMBER OF SWAP TRACKS?

At this point, you tell ETOS how much space is required for swapping. One swap track should be allocated for each 4K word area assigned. Therefore, if you have five 16K word jobs, enter 20. If you have four 20K jobs, enter 20. The system actually allocates 24 swap tracks, since four swap tracks are needed for the two system jobs - SCALE and DMON.

You should assign enough swap tracks to run your jobs, but you should not build in a great deal of extra tracks. Each track you assign takes away space from ETOS users. The swap tracks are created in an ETOS file called SWAPTRA.CKS. To calculate the size of this file, multiply (the number of specified swap tracks plus four) by thirty-two. This calculation gives you the number of blocks assigned to the swap file.

The distribution pack has eight swap tracks assigned. This

allows enough room for four users, each with an 8K word allocation. The size of the distribution swap file is 2000 (base 8) or 1024 (base 10) blocks. A sample creation of a swap file which will allows twelve users, each with a 16K word allocation, is shown in Figure 4-3.

Figure 4-3
Sample Creation of Swap Tracks

```
.R ETOS
ETOS V5B
OPTION? S
ARE YOU SURE? Y
NUMBER OF SWAP TRACKS? 48
OPTION?
```

The Sysgen option has another function in addition to creating the swap tracks. It creates an initialized ETOS file structure on the system pack. This file structure allows you to run a one-user ETOS from the system console. Only the system accounts [0,1], [0,2] and [0,3] (see 5.15) are created by this option. If you have created user accounts using the methods of Chapter 5, the Sysgen option would destroy them. You would have to recreate the user accounts after the Sysgen. If you wish to change the size of your swap file or your ETOS file structure becomes corrupted, you would use the Sysgen option at a later time. Otherwise, the Sysgen option is utilized only when you're installing ETOS the first time.

When you execute this option, you delete a number of accounts which contain reference information such as listings of the ETOS card reader, line printer and disk handlers. It is proper to delete this information, since you will not want this space wasted for daily operations. After you have been running the system for a period of time, you may wish to refer to this information. Section 5.16 lists the files on the original distribution pack. Since you might want to access these files,

you should insure that you perform the Sysgen on a copy of the distribution pack; not on the original itself.

4.3   TRANSFERRING VIRTUAL OS/8

If you have just performed a Sysgen, you must transfer virtual OS/8 to ETOS. This transfer will allow you to run OS/8 after you have LOGged IN to ETOS.

The transfer of virtual OS/8 is performed by copying the contents of the stand-alone OS/8 file ETOSET.SY to the ETOS file [0,2]OS8.RTS, which is created by Sysgen.   To perform this transfer, use the dialogue of Figure 4-4.

Figure 4-4
Transferring Virtual OS/8 to ETOS

```
.R ETOS              (not needed if Sysgen just performed)
ETOS V5B
OPTION? H
OPTION?
```

If you corrupt virtual OS/8 after you have been running ETOS for a period of time, you can shut down the system (see 4.7) and use the H option to restore virtual OS/8.

4.3.1   Transferring Virtual OS/8 System Programs

If you are installing ETOS from a distribution pack or a production pack, skip this subsection. Follow this subsection only if you built an ETOS pack from a non-disk medium, using the methods of 2.6.

Now that you have transferred an OS/8 system head (see 4.3) to virtual OS/8, you must transfer the general system programs.

These are the programs used by all ETOS users. All OS/8 programs

(e.g., PIP, PAL8) and COS programs (e.g., COMP, DFDIR) are contained in this virtual OS/8 area ([0,2] OS8.RTS). A list of these files is contained in 1.6 and 1.7. To transfer these files from a back-up device to virtual OS/8 SYS, use the procedure of Figure 4-5.

Figure 4-5
Transferring the ETOS Virtual OS/8 System Programs

```
.R ETOS                 (not needed if H option just performed)
ETOS V5B
OPTION? T
LOGIN;0,3 OPERATOR
JOB 03 LOGGED IN ON CONSOLE 00
?CANNOT FIND OS8DISK.DSK
.^VPRIV 4040
^VWE 0
```

At this point, mount the distribution medium marked ETOS VIRTUAL OS/8 #1 on the peripheral you built ETOS from. This peripheral is referred to as "dev" in the following write-ups. Use the following dialogue to copy all files from the backup unit to the virtual OS/8 storage area.

```
RUN  dev:PIP.El
*SYS:</Z/O
*^C
.RUN dev:FOTP.El
*SYS:<dev:
```

Depending on the user's software configuration, there may be several additional media marked "ETOS VIRTUAL OS/8 #X". For each of these units, follow the next procedure.

Dismount the unit currently on the backup peripheral. Mount the next sequential backup unit, write locked. Enter

```
*SYS:<dev:
```

Now, all programs are on the virtual OS/8 area. You should rename the system cusps from "El" extension to "SV" extensions and enable CCL commands by entering

Figure 4-5 (continued)

```
*SYS:.SV<SYS:*.E1/R
*^C
.R CCL
.^VS
!SHUTUP
```

You are now at the identical point which a user, who received a distribution pack from QUODATA, is at.   Follow the instructions presented in the rest of Chapter 4.

4.4   RETURNING TO STAND-ALONE OS/8

If you have initiated ETOS and you are waiting at the OPTION? question, you may wish to return to stand-alone OS/8.   This situation may occur if you remembered at this point that you hadn't backed up your disks with RKCOPY or SICOPY.   To return to stand-alone OS/8, use the dialogue of Figure 4-6.

Figure 4-6
Returning to Stand-alone OS/8

```
.R ETOS
ETOS V5B
OPTION? O
.                       (stand-alone OS/8 prints this)
```

4.5   STARTING ETOS FROM SINGLE-USER OS/8

To start ETOS, enter R ETOS<RET> to the keyboard monitor of the
03/04 console as seen in Figure 4-7, line (1)]. The terminal
responds by displaying OPTION? (line 2). Enter T<RET> or just
press <RET> to initiate time sharing. Any terminal may now be
LOGged IN (line 3).

Figure 4-7
Initiating ETOS Time Sharing

```
(1)    .R ETOS
       ETOS V5B
(2)    OPTION?T
(3)    LOGIN
       ETOS V5B AT 01:30:27 P.M. ON SUN 25-MAR-79
       ACCOUNT? 0,3
       PASSWORD? OPERATOR                        [NOT ECHOED]
       JOB 03 LOGGED IN ON CONSOLE 00
       ?OS8DISK.DSK NOT FOUND                    (warning message only)
       .
```

When time sharing is initiated via the T option, no initial
message is printed. However, a unique option pattern is
generated in the front panel lights. The octal number 5252 is
displayed in the accumulator. Then, the octal number 2525 is
displayed. The number 5252 is displayed and the pattern
continues. To see this pattern on a PDP8/E, turn the switch on
the front panel to AC. The pattern appears as a series of
rotating dots. To see this pattern on a PDP8/A with a
programmer's console, press the 4 button and then the DISP button
on the front panel. The pattern appears as a rotating sequence
of the numbers 5252 and 2525. On systems containing a limited
function panel, no display can be seen.

This pattern generated in front panel lights is called the null
job. The null job is run whenever no computation is being
performed by ETOS. When input/output is being buffered, this
buffering is treated as computation. When the actual

input/output is being performed, no ETOS overhead occurs. The null job is run at this time.

Figure 4-7 illustrated a LOGIN into account [00,03]. After you have just Sysgened a pack, this account is the only account you can LOGIN to. Chapter 5 will explain the concept of ETOS accounts and how to create them.

At this time, read Section 2.2 in the System User's Guide which provides you with additional information about the LOGIN process.

4.6   INITIALIZING THE SYSTEM TIME AND DATE

The ETOS time and date may be initialized in Figure 4-8 after ETOS is started and you have LOGged IN.

Figure 4-8
Sample Run of INIT

```
.R INIT

'INIT' V2.004

DATE (MM/DD/YY)? 01/30/77
TIME (HH:MM)? 13:10

.
```

The date entered to INIT is in the form of month/day/year. The time is entered in reference to a twenty-four hour clock, in the form of hours:minutes.

The job executing INIT should log out of ETOS after completion. Otherwise the user logged in while INIT was executed will have an incorrect run-time and connect-time.

The ETOS time of day is used by the OS/8 and COS keyboard monitors. Whenever the keyboard monitor is initialized and the

"." is printed, the COS or OS/8 date word is set equal to the ETOS date. Therefore, the date words are kept current if ETOS is left running overnight. The system time and date are also used by the programs SYSTAT and TIME, which run under OS/8. Figure 4-9 illustrates the printing of the time and date under OS/8.

Figure 4-9
Sample Use of the OS/8 Time and Date Commands

```
.TIME
01:11:02 P.M.
.DATE
FRIDAY, JANUARY 30, 1979
.
```

4.7 STOPPING ETOS

Hitting the HALT key is not recommended for stopping ETOS. If the computer is in the middle of a disk transfer when the HALT key is pressed, a bad block will be produced. This bad block causes problems the next time ETOS is started. The system can be stopped from any terminal by entering the dialogue of Figure 4-10.

Figure 4-10
Stopping ETOS

```
.^VS
!PRIV 4040
!SHUTUP

.
```

The system promptly responds by re-booting single user OS/8 on the 03/04 console. This command should be exercised with caution. Even though it insures that the current disk request is complete, it does not insure that the entire transfer is finished. If the user was in the middle of updating an OS/8 directory, it is possible that only a portion of the directory

would be written. This could result in a bad directory. It is recommended that the user utilize the BROADCAST command (see 6.2.3) to warn other users, a few minutes before shutting down the system.

If you are keeping track of cumulative job statistics with the USAGE program (see 6.4.14), insure that all users are LOGged OUT before you SHUTUP the system. These statistics are not updated until the user LOGs out. Any user who is not LOGged out will lose the statistics of his last terminal session. If you are not in a position to enter LOGOUT on all terminals, you may use the FORCE Command (see 6.2.4) to insure that all users have exited the system correctly.