CHAPTER 8
ERROR DIAGNOSIS

8.1 INTRODUCTION

The system manager is responsible for the operation of ETOS. The previous chapters deal with what you do with standard ETOS processing when everything is going right. This chapter discusses non-standard processing off of non-system drives, diagnosing ETOS problems, crash recovery and bug reporting and resolution. Since computer processing often follows Murphy's Law, it is as important to know what to do when things go wrong as it is to know what to do when things go right.

8.2 ADDITION AND REMOVAL OF TERMINALS

The program TTYSET provides an easy method for physically removing or adding a terminal to ETOS, without shutting the system down. This change of terminals assumes that you are plugging into or unplugging from a cable which is attached to the interface. No interface changes may be made without shutting the system down. TTYSET can also be used to reactivate a terminal which is not responding, free a console slot for a detached job or disable a user who is abusing the system. All changes made with TTYSET are lost when ETOS is shut down. Permanent terminal configuration changes are made with the CONFIG program (see 3-9).

To run TTYSET, LOGIN to a privileged account and enter R TTYSET<RET>. The system responds with a dollar sign ($). At this point, you can enter one of the valid TTYSET commands. After all commands have been specified, enter Q to exit the program.

Table 8-1 contains a list of TTYSET commands and their meanings.

Table 8-1
TTYSET Commands

Command	Meaning
B n[,m]	To print a console's current BREAK mask (see 3.7.5, System User's Guide), enter B n, where "n" is the console number. To change the BREAK mask, enter B n,m, where "n" is the console number and "m" is the new BREAK mask.
E n,m[,o]	This command enables terminals which were killed or were not specified in the CONFIG program. To eanble a terminal, which runs off of a KL8-E, KL8-JA or DKC8-AA, enter E n,m, where "n" is the console number and "m" is the teleprinter code of the terminal interface. To enable a terminal, which runs from a KL8-A, enter E n,m,o, where "n" is the console number, "m" is the device code of the KL8A and "o" is the KL8A line number.
K n	Disable console "n". Any characters typed on this console are ignored after this command is entered. Do not disable your own console.
L	Lists all consoles, their teleprinter code and their line numbers (for KL8As). All KL8As are noted with an asterisk (*). This information is obtained from the IOT TABle (see 3.9). Any console number which has an entry of Ø in the IOT TABle does not have a terminal associated with it. These dummy consoles are used for DETACHED jobs. The DETACH command disassociates the current job with the original console number and associates it with the first available dummy console. Consoles being used for DETACHED jobs are indicated on the listing.
R n	Rescues console "n", which is not responding to any terminal input. The console's KSTAT word is set to 7377 and the BREAK mask is set to 4ØØØ. If this command does not reactivate a console, shut down the system and reinitialize to activiate it.
S n[,m]	To print a console's KSTAT word (see 3.7.6, System User's Guide), enter S n, where "n" is the console number. To change the KSTAT word, enter S n,m, where "n" is the console number and "m" is the new KSTAT value.

Figure 8-1 contains a sample run of TTYSET.

Figure 8-1
Sample Run of TTYSET

```
.R TTYSET
'TTYSET' V2.025
$ L
                    DEVICE              LINE
CONSOLE              CODE               NUMBER
  Ø                   Ø4
  1                   4Ø                  Ø
  2                   4Ø                  1
  3                   4Ø                  2
  4                   4Ø                  3
  5                  DETACHED
  6                  DUMMY
  7                  DUMMY
 1Ø                  DUMMY
 11                  DUMMY
 12                  DUMMY
 13                  DUMMY
 14                  DUMMY
 15                  DUMMY
 16                  DUMMY
 17                  DUMMY

$E 6,47             (enable KL8JA board with output code 47)
$K 4                (kill console 4)
$L
                    DEVICE              LINE
CONSOLE              CODE               NUMBER
  Ø                   Ø4
  1                   4Ø                  Ø
  2                   4Ø                  1
  3                   4Ø                  2
  4                  DUMMY
  5                  DETACHED
  6                   47
  7                  DUMMY
 1Ø                  DUMMY
 11                  DUMMY
 12                  DUMMY
 13                  DUMMY
 14                  DUMMY
 15                  DUMMY
 16                  DUMMY
 17                  DUMMY
$Q
.
```

When TTYSET is run, it sets your PRIV word to 4040. When TTYSET is exited, it sets your PRIV word to 0040. A list of all possible TTYSET error messages follows.

?BAD ARG

A console number greater than 17 has been entered as an argument to one of the commands

?BAD IOT TABLE - PROGRAM INCOMPATABILITY

An entry in the IOT table is not in the form 6XX5 or 0XXY. The IOT tabel is verified in TTYSET initialization. If this error is printed, you must SHUTUP the system and run CONFIG (see 3.9) to fix the IOT TABle error.

?INACTIVE CONSOLE

An inactive (dummy) console was entered as an argument to a B,K,R or S command.

?INVALID ENTRY

When listing the IOT table, an entry was found, which was not in the form 6XX5 or 0XXY. This error is usually detected on TTYSET initialization. The only time this error is printed is if the IOT table is changed via the POKE command (see 6.2.2) after the TTYSET program was started. If this error is printed, you must SHUTUP the system and reinitialize ETOS.

?PROTECTION VIOLATION

The TTYSET program was run from a non-privileged account. You must LOGOUT of the current account, LOGIN to a privileged account and run TTYSET.

?SYNTAX

A non-octal digit was typed as an argument or two many arguments were specified for one of the TTYSET commands

?UNKNOWN COMMAND

A command was specified which was not B, E, K, L, R or S.

8.3 RUNNING FROM NON-SYSTEM DRIVES

For System Industries ETOS installations, ETOS always runs from the fixed disk in port Ø. There is no way to change this procedure. For RKØ5 ETOS installations, ETOS usually runs from the RKØ5 pack in drive Ø. It is possible to modify ETOS to run on an RKØ5 pack in drive 1, 2 or 3. This is normally done only when drive Ø is not operating due to a hardware problem. A general procedure for modifying ETOS to run on any drive follows.

1) Bootstrap an ETOS system pack in the drive you wish to run from. If this drive number is Ø, use the procedures outlined in 2.2. If this drive number is 1, 2 or 3, use the procedure outlined in the OS/8 Handbook, page 1-27. If you do not have a front panel, you must bootstrap OS/8 in another peripheral and type in the multi-drive RKØ5 bootstrap with ODT.

2) Enter the commands of Figure 8-2 under stand-alone OS/8.

Figure 8-2
Running ETOS from Non-System Drives

```
.R CONFIG
'CONFIG' V3.10
*RMON,0564=70XX
*RMON,11633=640Y
*RMON,12640=640Y
*RMON,12730=640Y
*DMON,00037=640Y
*EXIT
.
```

In the above writeup, "XX" is equal to the drive number plus Ø6. For drive Ø, XX = Ø6; for drive 1, XX = Ø7; for drive 2, XX = 1Ø; for drive 3, XX = 11. "Y" is equal to the drive number multiplied by 2. For drive Ø, Y = Ø; for drive 1, Y = 2; for drive 2, Y = 4; for drive 3, Y = 6. For example, the commands of Figure 8-3 can be used to modify ETOS to run from drive 1.

Figure 8-3
Sample Run of ETOS from a Non-System Drive

```
.R CONFIG
'CONFIG' V3.1Ø
*RMON,Ø5643=7ØØ7
RMON (Ø5643) 7ØØ6 7ØØ7
*RMON,11633=64Ø2
RMON (11633) 64ØØ 64Ø2
*RMON,12640=64Ø2
RMON (12640) 64ØØ 64Ø2
*RMON,12730=64Ø2
RMON (12730) 64ØØ 64Ø2
*DMON,ØØØ37=64Ø2
DMON (ØØØ37) 64ØØ 64Ø2
*EXIT
.
```

3) Initiate ETOS in the normal manner by entering R ETOS <RET>. Answer the "OPTION?" question and ETOS is up and running.

When you wish to terminate ETOS processing, you normally use the SHUTUP command (see 4.7). If you are running from drive Ø, the SHUTUP command cannot be used. SHUTUP always bootstraps drive Ø. To terminate ETOS on drives 1, 2 or 3, run the SHUTUP program by LOGging INto a privileged account and enter R SHUTUP<RET>. The system responds with "DRIVE NUMBER?".

Enter Ø, 1, 2 or 3 to boot an OS/8 system pack in the specified drive. SHUTUP sets your PRIV word to 4Ø4Ø. A sample run is given in Figure 8-4.

Figure 8-4
Sample SHUTUP of ETOS from a Non-system Drive

```
.R SHUTUP
'SHUTUP' V1.ØØØ7
DRIVE NUMBER? 1
.                          (Stand-alone OS/8 is now
                            running on drive 1)
```

8.4 DIAGNOSING DISK ERRORS

If the system is stopping in the middle of ETOS processing, there must be a software or hardware problem. The hardware may be physically installed improperly or the software may be configured improperly. There also might be a hardware or software malfunction. Sections 8.4 through 8.8 contain information which assist you in determing the source of the problem. One possible source of system malfunction is hardware errors on the RKØ5 or System Industries disks. Sections 8.4.1 and 8.4.2 contain information which will assist you in the diagnosis of disk errors.

8.4.1 RKØ5 Disk Errors

Whenever an errors occurs on a disk, ETOS logs the error in an internal buffer. If an abnormal number of errors occur, the buffer will fill up and ETOS will crash. The ERRCPY program prints out all errors which have occurred since system start-up or since the previous running of the program. It also empties the buffer area so that the same errors will not print the next time the program is run. To run ERRCPY, enter R ERRCPY<RET>. All errors are displayed at this time. If no errors are in the buffer, the system prints "DONE" and exits.

You must be careful when you interpret the output of ERRCPY. There are some processes which will cause an error, even though the error is an operational error, not a physical device error. For example, the sequence of commands of Figure 8-5 will cause an error if disk drive 1 is not on-line.

Figure 8-5
Sample Operational RKØ5 Disk Error

```
.^VS
!ASSIGN DK1:
!LOOKUP 3=DK1:
!BOOT
.DIR CHN3:
BAD DIRECTORY
.
```

ERRCPY sets your PRIV word to 4040 upon entry. After all errors are printed, your PRIV word is set to 0040. A sample run of ERRCPY is given in Figure 8-6.

Figure 8-6
Sample Run of ERRCPY

```
.R ERRCPY
'ERRCPY' V2.006
RKØ5 ERROR
ERROR = WRITE LOCK ERROR + DRIVE STATUS (0022)
COMMAND REG. = WRITE DATA, INT. ON DONE
FIELD = 1, DRIVE 1, SECTOR = 00001, CURRENT ADDRESS = 6200
.
```

The format of the errors is similar to the errors printed out by the RKØ5 hardware diagnostic. Refer to this diagnostic or the RKØ5 Maintenance Manual to interpret the output of ERRCPY.

Whenever the system is initiated via R ETOS, all previous errors are lost.

8.4.2 System Industries Disk Errors

To diagnose hardware problems with a System Industries disk drive, boot an OS/8 mass storage medium which contains the 3040 program. The ETOS distribution pack or the fixed operational pack can be booted in port 0 (using the procedures of 2.2.3) since the 3040 program is on those packs. The 3040 program is used to format disks (see 2.4.2) and diagnose disk errors. 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 diagnose hardware problems with 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 blank disk or disks, which you wish to use as scratch media 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 yu 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 test 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 tests the fixed disk. If "X" equals "5", the system tests the removable disk. If "X" equals "7", the system tests both fixed and removable disks. For example, if you want to test 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, which you wish to test are set to run and that the READY light is on. Enter 0040 to initiate formatting and diagnosis.

7) When all testing is complete, the system prints "PASS 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 given in Figure 8-7.

Figure 8-7
Sample Run of the 3040 Program

```
.R 3040

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

PASS COMPLETE
.
```

8.4.3 Verifying Bad Disk Blocks

When the RK05 or System Industries disk hardware writes a block, it automatically calculates a checksum. This checksum is the truncated total of all of the words in the block. This checksum is written at the end of the block. When a block is read, the checksum is inspected to make sure that it is the total of the words in the block. This verification process is known as a cyclic redundancy check (CRC). Errors found during this process are known as CRC errors. If the software reads a block with a bad checksum, a read error, halt or another unpredicatable event occurs. CRC errors are produced when a write operation is interrupted in the middle of a block, due to a power failure, a crash or injudicious use of the halt key. You should always shut ETOS down via the SHUTUP command (see 6.2.7) to insure that CRC errors are not produced when ETOS processing is terminated.

The OS/8 program, FUTIL, has the capability to search a device for bad blocks (blocks with CRC errors). The IMAGE files (see

5.15.2) are used to accomplish this. The file IMAGE.RK5, which is used for the DKCOPY program (see 6.6), is a dummy file which allows you to access any block in the system disk. Without this file, ETOS would prohibit you from accessing some of the ETOS system files (e.g., map blocks). IMAGE.RK5 cannot be used with FUTIL, because FUTIL cannot handle devices larger than 4096 (decimal) blocks. Therefore, smaller IMAGE files have been created to map smaller sections of the disk. On an RK05, IMAGE.RKA is used to inspect blocks 0 to 6257. IMAGE.RKB is used for blocks 6260 to 14537. On a System Industries disk, IMAGE.RKA maps blocks 0 to 6277. IMAGE.RKB maps blocks 6300 to 145777. IMAGE.RKC maps blocks 14600 to 23077. IMAGE.RKD maps blocks 23100 to 31377.

To inspect a section of the system device for lead errors, LOOKup an IMAGE file on a channel and use the FUTIL Command SCAN (see 3.2.8) to perform the verification. A sample inspection of blocks 0 to 6257 on an RK05 system is contained in Figure 8-8.

Figure 8-8
Sample Verification of Bad Blocks

```
.^VS
!CLOSE 3
!PRIV 4040
!LOOKUP 3 = [0,2]IMAGE.RKA
!WE 3
!CONTINUE
R FUTIL
SET DEVICE CHN3
SCAN 0-6257

EXIT
.
```

8.5 MFD AND UFD STRUCTURE

Section 5.2 contains documentation on the ETOS file structure. That section must be understood completely before section 8.5 is

read. This section contains system file layouts which will help you to understand the ETOS directory structure more fully.

After a Sysgen, all of the disk not assigned to ETOS system files is available for allocation to accounts and ETOS files. There are several major differences between allocation of ETOS files and allocation of OS/8 or COS files. Under OS/8 or COS, yu can only have one user at a time modifying a directory of a storage area. Under ETOS, any number of users can modify the ETOS directory by adding or deleting accounts and ETOS files. OS/8 or COS can only handle files less than 49Ø95 blocks. ETOS uses two 12-bit words for file length instead of one 12-bit word, so that ETOS files can be up to 8,388,6Ø8 blocks long.

The last major difference in directory handling is that OS/8 and COS allocate only contiguous files. When a file is deleted, an empty is created. If you have a six block empty and a five block empty, you cannot create an eleven block file. Periodically, you must use PIP, option S under OS/8 or PIP option E under COS to consolidate all empty areas into one contiguous empty. ETOS supports two types of files. Fixed length files are similar to OS/8 or COS files in that they are contiguous files. They cannot be allocated unless there is enough contiguous space for the entire file. The other type of file is a variable length file. This type of file may be segmented. For example, if you had three empties of four blocks, a twelve block variable length file can be allocated. As a user, you do not need to be aware of how the file is segmented. ETOS takes care of this segmentation automatically. For example, if you use the READ CHANIO command, block numbers are logical rather than physical. Relative block number thirty might be in the third segment of the file on physical disk block number 5ØØØ. This mapping between logical and physical blocks is transparent to you. the only file where logical and physical block numbers are the same is the IMAGE.RK5 file.

To support variable length segmented files, ETOS uses the concept of a map block. The map block contains a list of segments which make up the file. The format of the map block is contained in Table 8-2.

Table 8-2
Map Block File Format

Locations (octal)	Meaning
0-1	These two words make up a double precision number which represents the starting physical block of the first segment. Location 1 is the high order portion of the block number.
2-3	These two words make up a double precision number which represents the ending physical block of the first segment. Location 3 is the high order portion of the block number.
4-5	Starting block number of second segment.
6-7	Ending block number of second segment.
.	
.	
.	
374-375	Starting block number of sixty-fourth segment.
376-377	Ending block number of sixty-fourth segment.

A variable length file does not usually contain sixty-four segments. Therefore, a starting segement block number of 00000000 indicates the end of the segment definitions. A sample map block for a thirty (octal) block file segmented into three parts is contained in Table 8-3.

Table 8-3
Sample Map Block

Locations	Contents	Description
Ø	6ØØØ	Starting block at first
1	ØØØØ	segment = ØØØØ6ØØØ
2	6ØØ4	Ending block of first
3	ØØØØ	segment = ØØØØ6ØØ5
4	74ØØ	Starting block of second
5	ØØØØ	segment = ØØØØ74ØØ
6	742Ø	Ending block of second
7	ØØØØ	segment = ØØØØ742Ø
1Ø	Ø1Ø5	Starting block of third
11	ØØØ1	segment = ØØØ1Ø1Ø5
12	Ø1Ø7	Ending block of third
13	ØØØ1	segment = ØØØ1Ø1Ø7
14-377	ØØØØ	End of segments

The master file directory is basically a list of all accounts on the disk. In addition to the account attributes (see 5.2), each account entry contains a pointer to the map block which defines the layout of the user file directory for the account. The master file directory is contained in a variable length ETOS file. This file can be any number of blocks. Sixteen account entries are contained in each block. For format of a master file directory block is contained in Table 8-4.

Table 8-4
Master File Directory Block Format

Locations	Meaning
Ø	Account number for first account in the block. The first six bits of the word contain the project number. The last six bits of the word contain the programmer number.
1-4	Account password. The password can be from one to eight characters. Each word contains two six-bit ASCII characters. If the password is less than eight characters, the remaining characters are equal to ØØ.
5	Account protection code.
6	Account creation date. The format of this word is the same as the date word contained in Table 7-16, System User's Guide.
7-1Ø	Double precision complement of the location of map block, which defines the user file directory for this account. Location 7 is the low order portion of the pointer to the map block.
11-12	Double precision length of the user file directory. Location 11 is the low order portion of the length.
13-17	Unused. Set to ØØØØ.
2Ø-37	Account entries for second account in the block.
.	
.	
.	

An account number of ØØØØ indicates a deleted account. An account number of ØØØØ, with a password block, starting with 7777 and ØØØØ, indicates the end of master file directory entries. A sample master file directory block which contains account [Ø,17] with a password of TEST, account [17,2Ø] with a password of OS8USE is contained in Table 8-5.

Table 8-5
Sample Master File Directory Block

Locations	Contents	Description
0	0017	First account number
1	2405	TE (password)
2	2324	ST
3	0000	
4	0000	
5	0077	Protection code
6	2011	Creation date (April 1, 1979)
7	4750	Low order map block location
10	0000	High order map block location
11	0001	Low order UFD length
12	0000	High order UFD length
13-17	0000	Unused
20	1720	Second account number
21	1723	OS (password)
22	7025	8U
23	2305	SE
24	0000	
25	6077	Protection Code
26	2571	Creation date (May 17, 1979)
27	6744	Low order map block location
30	7777	High order map block location
31	0002	Low order UFD length
32	0000	High order UFD length
33-37	0000	Unused
40-377	0000	End of directory entries

Each account on the system has a user file directory which contains a list of all ETOS files under the account. In addition to the file attributes (see 5.2), each variable length entry contains a pointer to the map block which defines the layout of the ETOS file. Each user file directory is contained in a variable length ETOS file. This file can be any number of blocks. Sixteen file entries are contained in each block. The format of a user file directory block is contained in Table 8-6.

Table 8-6
User File Directory Block Format

Locations	Meaning
0-4	ETOS file name for the first file on the block. The file name consists of a name and extension. The name is from one to seven characters. The extension is from one to three characters. If the name or extension is less than the maximu, the remaining characters are equal to 00.
5	File protection code.
6	File creation date.
7-10	Double precision complement of the location of map block, which defines the layout of a variable length ETOS file. Location 7 is the low order portion of the pointer to the map block. For fixed length files, these locations are a double precision pointer to the starting block of the ETOS file.
11-12	Double precision length of the ETOS file. Location 11 is the low order portion of the length.
13-17	Unused. Set to 0000.
20-37	File entries for second file in the block.
.	
.	
.	
360-377	File entries for sixteenth file in the block.

A file name which is set to 0000 indicates a deleted ETOS file. A file name of 0000 and a filename block, starting with 7777 and 0000, indicates the end of user file directory entries. A sample user file directory block, which contains a variable length OS/8 storage area (OS8DISK.DSK) of 200 (octal) blocks and a fixed length COS program storage area (COSDISK.RTS) of 547 (octal) blocks, is contained in Table 8-7.

Table 8-7
Sample User File Directory Block

Locations	Contents	Description
Ø	1723	OS (ETOS filename)
1	7ØØ4	8D
2	1123	1S
3	13Ø4	KD
4	2313	SK
5	ØØ5Ø	File protection code
6	3311	Creation date (June 25, 1979)
7	57Ø3	Low order map block pointer
1Ø	7777	High order map block pointer
11	Ø2ØØ	Low order file length
12	ØØØØ	High order file length
13-17	ØØØØ	Unused
2Ø	Ø317	CO (ETOS filename)
21	23Ø4	SD
22	1123	IS
23	1322	KR
24	2423	TS
25	ØØ74	File protection code
26	3311	Creation date (June 25, 1979)
27	ØØØØ	High order file pointer
3Ø		Low order file pointer
31	Ø547	Low order file length
32	ØØØØ	High order file length
33-37	ØØØØ	Unused
4Ø-377	ØØØØ	End of directory entries

There is one additional ETOS file which is used in conjunction with the master file directory, user file directory and map blocks to allocate disk space. This file called FREEBLO.CKS is stored in account [Ø,2]. When blocks are assigned by the system to the MFD, UFD and map blocks or by the user to eTOS files, the FREEBLO.CKS file tells ETOS where to allocate blocks. The FREEBLO.CKS contains a list of the starting physical block and ending block of all empty areas on the disk. When a segment is allocated, the block numbers are removed from the FREEBLO.CKS file and inserted into the MFD, UFD on map block files. Section 8.6 contains documentation about printing the contents of the FREEBLO.CKS file. The format of a block of this file is contained in Table 8-8.

Table 8-8
FREEBLO.CKS File Block Format

Locations	Meaning
Ø-1	Double precision starting block number of first empty area listed in this block. Location Ø contains the low order portion of this number.
2-3	Double precision ending block number of first empty area listed in this block. Location 2 contains the low order portion of this number.
4-5	Starting block number of second empty listed in this block.
6-7	Ending block number of second empty listed in this block.
.	
.	
.	
374-375	Starting block number of sixty-fourth empty listed in this block.
376-377	Ending block number of sixty-fourth empty listed in this block.

There is not sixty-four entries in each FREEBLO.CKS block. A starting block number of ØØØØØØØØ is used to terminate the list of empties. A sample FREEBLO.CKS block containing seven entries is contained in Table 8-9.

Table 8-9
Sample FREEBLO.CKS File Block

Locations	Contents	Description
0	4556	Low order starting block
1	0000	High order starting block
2	4557	Low order ending block
3	0000	High order ending block
4	4572	Low order starting block
5	0000	High order starting block
6	4603	Low order ending block
7	0000	High order ending block
10	4605	Low order starting block
11	0000	High order starting block
12	4605	Low order ending block
13	0000	High order ending block
14	4612	Low order starting block
15	0000	High order starting block
16	4630	Low order ending block
17	0000	High order ending block
20	7062	Low order starting block
21	0000	High order starting block
22	7167	Low order ending block
23	0000	High order ending block
24	4300	Low order starting block
25	0001	High order starting block
26	4377	Low order ending block
27	0001	High order ending block
30	4401	Low order starting block
31	0001	High order starting block
32	4537	Low order ending block
33	0001	High order ending block
34-377	0000	Empty termination

In ETOS file which are used as OS/8 storage areas (OS8DISK.DSK) or COS program storage areas (COSDISK.RTS), there is a directory of OS/8 programs and files or COS programs. The ETOS file handler does not know about these directories. All directory mamipulations are performed by OS/8 or COS. If is for this reason that the stand-alone OS/8 or COS limitations (i.e., file length less than 4095 blocks) exist under ETOS. The OS/8 and COS directory structures are identical except that COS does not store files in the directory only COS programs are contained in the directory. The format of an OS/8 or COS directory is contained in Appendix A in the OS/8 Software Support Manual. It is

important to realize that free blocks or empties mentioned in the OS/8 manual are completely separate from the ETOS list of free blocks. Once an OS/8 or COS storage area is allocated, all of the blocks in that area are removed from the ETOS list of free blocks. A DIR or DI command might list free blocks in an OS/8 or COS area. These blocks are not part of the ETOS free blocks because they are not available to be assigned to other ETOS files. They are only available for use by the owner of the storage area.

8.6 PRINTING FREE BLOCKS FILE

The ETOS file [Ø,2] FREEBLO.CKS contains a list of all blocks which have not been allocated to ETOS files. The format of this FREEBLO.CKS file is documented in section 8.5. To print the the contents of this file, the program FREE may be used. To run this program, LOGIN to a privileged account and enter R FREE<RET>. The system responds with "DK NUMBER?". Enter Ø to inspect the system pack or 1, 2, or 3 to inspect public packs DK1, DK2 or DK3. The starting octal block, ending octal block and octal length of every free area on the ETOS system or public pack is printed. At the end of the list, the total number of free blocks is printed in octal and decimal. If you suspect a corruption of the ETOS file structure, inspect the free areas closely. If any of the areas overlap, your file structure has been corrupted. A Sysgen should be performed to recreate the ETOS file structure (see 4.2). If you use FREE to inspect a non-system public pack, the pack must be MOUNTed (see 6.2.6). A sample run of the FREE program is given in Figure 8-9.

Figure 8-9
Sample Run of FREE

```
.R FREE
'FREE' V1.Ø1Ø

DK NUMBER? Ø

START           END                LENGTH

00004556        00004557           00000002
00004572        00004603           00000012
00004605        00004605           00000001
00004612        00004630           00000017
00007062        00007167           00000106
00014300        00014377           00000100
00014401        00014537           00000137

TOTAL FREE BLOCKS:                 00000405      261

.
```

FREE sets your PRIV word to 4040, closes channel 7 and LOOKs UP the FREEBLO.CKS file on channel 7. When the listing is finished, channel 7 is closed and the PRIV word is set to 0040.

A list of all possible FREE error messages follows.

?X LOOKUP ERROR

A lookup error occurred while attempting to look up the FREEBLO.CKS file. "X" corresponds to the errors in the LOOKUP CHANIO instruction (see 7.4.6, System User's Guide). If this error occurs, your data structure is corrupt on the pack you specified. You must execute the Sysgen option of ETOS (see 4.2) if the pack is DKØ or the DSKINT program (see 5.11) if the pack is DK1, DK2 or DK3. The Sysgen option or the DSKINT program recreate the ETOS file structure.

?PROTECTION VIOLATION

The FREE program was run from a non-privileged account. You must

LOGOUT of the current account, LOGIN to a privileged accou run FREE.

?X READ ERROR

A READ error occurred while READing the FREEBLO.CKS file. "X" corresponds to the errors in the READW CHANIO instruction (see 7.4.1, System User's Guide). If this error occurs, your data structure is corrupt on the pack you specified. You must execute the Sysgen option of ETOS (see 4.2) if the pack is DKØ or the DSKINT program (see 5.11) if the pack is DK1, DK2, or DK3. The Sysgen option or the DSKINT program recreate the ETOS file structure.

8.7 RECOVERY AFTER A CRASH

In the event that ETOS is crashing repeatedly, the following procedure should be executed immediately after a crash.

1. Write protect all disks.
2. Halt the physical machine.
3. Set the switch register to 7600. On a PDP8/E, this is accomplished by setting switches Ø-4 up and 5-11 down. On a PDP8/A, this is accomplished by pressing the 7, the 6, the Ø and the Ø numeric buttons and then pressing the LSR button. If you do not have a programmer's console, you cannot save a crash dump.
4. Initiate stand-alone OS/8. On a PDP8/E, this is accomplished by pressing the ADDR LOAD, EXTD ADDR LOAD, CLEAR and CONT keys in succession. On a PDP8/A, this is accomplished by pressing the LA, LXA, INIT and RUN buttons in succession.
5. The OS/8 prompter dot should appear on the console terminal. Enter the following.

   .SAVE dev:CRASHn 20000-27777,3000-37777;7600

   where "dev" is any OS/8 file structured device, "CRASHn" is any unique file name, for example "CRASH1" or "CRASH2".

   You can SAVE the crash dump file on stand-alone SYS if there are 33 contiguous free blocks. if there are not

33 contiguous free blocks, you must mount a pack with free space in a non-system drive or another type of OS/8 peripheral (e.g., RXØ1, TD8E).

6. Save all terminal output and record the circumstances surrounding the crash. Make sure that QUODATA receives this material along with the CDUMP listing or crash files.

This procedure creates the crash file "dev:CRASHn.SV". CDUMP is a program that prepares octal dumps of saved ETOS crash files. These dumps are customized for ETOS to facilitate easy analysis. Part of the crash file is compared with ETOS.SV to determine which locations have changed. To initiate CDUMP, enter R CDUMP<RET>. The program responds with an asterisk (*). At this time, you specify an output print device and your crash files in the form

*output:<input:ETOS.SV,crash:CRASH1.SV[,CRASH2.SV,...]

CDUMP can operate on up to eight crash files, generating one output listing that compares each crash with the other crashes and the original form of ETOS. It is generally preferable to dump as many crashes as possible per run. CDUMP can run under ETOS or stand-alone OS/8. A sample crash dump procedure performed under stand-alone OS/8 with crash files saved on an RXØ1 floppy disk are given in Figure 8-1Ø.

Figure 8-1Ø
Sample Crash Dump Under Stand-alone OS/8

```
.SAVE RXAØ:CRASH4 2ØØØØ-27777,3ØØØØ-37777;76ØØ
.R CDUMP
'CDUMP' V5.ØØ2
*LPT:<SYS:ETOS,RXAØ:CRASH1,CRASH2,CRASH3,CRASH4
.
```

A sample crash dump performed under ETOS, with crash files saved on stand-alone OS/8 sys, is listed in Figure 8-11.

Figure 8-11
Sample Crash Dump Under ETOS

```
.SAVE SYS:CRASH2 20000-27777,30000-37777;7600
.R ETOS
ETOS V5B
OPTION? T
H;0,3 OPERATOR

JOB 03 LOGGED IN ON CONSOLE 00
.^VS
!ASSIGN LPT:
!LOOKUP 3=[0,2]OS8.OS8
!CONTINUE
RUN CHN3:CDUMP
'CDUMP' V5.002
*LPT:<CHN3:ETOS,CRASH,CRASH2
.
```

A sample crash dump performed under ETOS, with crash files saved on both portions of an OS/8 pack mounted in RK05 drive 1 is given in Figure 8-12.

Figure 8-12
Sample Crash Dump Under ETOS

```
.SAVE RKA1:CRASH3 20000-27777,30000-37777;7600
.R ETOS
ETOS V5B
OPTION? T
H;0,3 OPERATOR

JOB 03 LOGGED IN ON CONSOLE 00
.^VS
!ASSIGN LPT:
!LOOKUP 3=[0,2]OS8.OS8
!ASSIGN DK1:
!LOOKUP 4=DK1:
!LOOKUP 5=DK1:, 6260
!CONTINUE
RUN CHN3: CDUMP
'CDUMP' V5.002
*LPT: <CHN3:ETOS, CHN4: CRASH1, CRASH2, CHN5: CRASH3
.
```

Figure 8-13 contains a sample crash dump performed under ETOS, with crash files on a TD8E DECtape.

Figure 8-13
Sample Crash Dump Under ETOS

```
.SAVE DTA0:CRASH4 20000-27777,30000-37777;7600
.R ETOS
ETOS V5B
OPTION? T
H; 0,3 OPERATOR

JOB 03 LOGGED IN ON CONSOLE 00

.^VS
!ASSIGN LPT:
!LOOKUP 3=[0,2]OS8.OS8
!PRIV 4040
!CONTINUE
RUN CHN3:CDUMP
'CDUMP' V5.002
*LPT:<CHN3:ETOS,DTA0:CRASH1,CRASH2,CRASH3,CRASH 4
.
```

The previous examples output crash dump listings to the line printer. The output specification may be any OS/8 file. If no line printer is available, TTY: or LT: may be used to obtain a listing on the terminal. You do not have to ASSIGN the TTY.

The first input specification must be the file ETOS.SV that was running when the crash occurred. One to eight crash files must also be specified for input.

8.8 DEBUGGING THE MONITOR

If you are having serious problems with ETOS, QUODATA may request that you run a version of ETOS which has a debugging tool built in. This debugging tool (XDDT7), which is stored on stand-alone OS/8 SYS, uses the upper 4K words of physical memory. Since 8K words of user memory are required and 8K words is taken up by the ETOS resident monitor, this procedure cannot be performed on a

machine which contains less than 32K words of memory. To create a debugging version of the monitor, use the procedure of Figure 8-14 under stand-alone OS/8.

Figure 8-14
Creating a Debugging Version of ETOS

```
.R CONFIG
'CONFIG' V3.012
*MEMTAB,7=1
MEMTAB(00007) 0000 0001
*EXIT
.R PIP
*SYS:XETOS.SV<SYS:XDDT.SV,ETOS.SV/I
*^C
.
```

After this debugging monitor is saved, you can run it by entering the commands of Figure 8-15.

Figure 8-15
Sample Execution of Debugging Version of ETOS

```
.R XETOS
1#
4000' ETOS V5B
OPTION? T

JOB 03 LOGGED IN ON CONSOLE 00
.
```

8.9 EXAMINING PHYSICAL MEMORY

With the PEEK command (see 6.2.2), you can examine the contents of physical memory. However, if the contents of the locations which you are examining are changing rapidly, you cannot keep up with the changes. If you have a front panel, you can watch changes being made to addresses in physical memory. The front panel is normally set to display the accumulator so you can see the null job and determine the use of ETOS. However, the examination of memory requires that you display the Multiplier

Quotient (MQ). To display the MQ on a PDP8/E, turn the rotary switch on the front panel to the position labeled MQ and insure that the key switch is set to power, not panel lock. To display the MQ on a PDP8/A with a KC8-AA programmer's console, press the "1" button and then the DISP button on the front panel. To set the field, which you wish to examine, set bits Ø-8 to zero. On a PDP8E, this is accomplished by setting switches Ø-8 down. On a PDP8/A, this is accomplished by pressing the "Ø" button on the front panel three times. When bits Ø-8 have been set to Ø, you can set bits 9-11 to the field which you wish to examine. On a PDP8/E, this is accomplished by treating switches 9-11 as an octal number in the range Ø-7. On a PDP8/A, press the number button corresponding to the field you wish to examine and press the LSR button. Now that the field is set, set the switch register to the address you wish to examine. On a PDP8/E this is accomplished by setting the switches to the twelve-bit address you wish to examine. On a PDP8/A, press four numeric buttons to represent the four octal numbers in the address and press the LSR button.

To change the address you are examining, change the switch register. To change the field you are examining, set bits Ø-8 to zero and bits 9-11 to the desired field. You cannot examine locations Ø-7 in any field, since this requires setting bits Ø-8 to zero, which resets the field.

8.1Ø SOFTWARE PERFORMANCE REPORTS

If you have a problem with one of the ETOS components, fill out a software performance report. These forms are contained in the release notes. When we receive the software performance report, we will respond in writing within thirty days. The response could be a patch, an explanation or an acknowledgement of the problem with a time frame for its solution. Written communication is essential for proper support. We also request

that you write to us about new features which you would like to see in the next version. Most of the features of a new release are decided by tabulating these customer requests.

The Software Performance Report is your best method of communicating with us. Our method of communicating with you is the technical summaries. These summaries are discussed in the next section.

8.11 TECHNICAL SUMMARIES

When we begin to work on a problem with one of the ETOS components, we consult any program discrepancy forms which deal with the problem. If more information is required, we contact the customers who sent in the forms. The next step in the debugging process is the recreation of the problem at QUODATA. Once we can reproduce the problem, the only step remaining is to devise the best method for the problem's solution. This solution is normally a modification to the ETOS software. However, it may be an operational solution, which requires a different method of processing. This solution is implemented in our development system. After the solution is tested, we test the solution further by having a customer implement it into his system. After we are confident that the solution is valid, we distribute it to all users via the technical summaries.

Technical summaries are periodic mailings from QUODATA, which allow us to support a large number of customers in the most effective manner. These summaries contain solutions to customer problems, which are devised by the methods discussed previously. The summaries also contain documentation changes, notification of new products and general information about ETOS.

The information of the technical summaries may be communicated through technical summary letters, by documentation page

replacements, by optional patches and by mandatory patches. Besides containing general information about ETOS and notification of new products, the technical summary letters (see Appendix D) can also act as a forum for user discussions. If you have a program of general interest, send us a description of the program. We will include the description and your installation name and address in the next technical summary. Any users who are interested in the program can contact you directly. QUODATA does not distribute user software. You should also write to us if you require a program to perform a specified task. We will include your request in the next technical summary.

Documentation are distributed in the form of replacement pages. We distribute the manuals in three-ring binders. New documentation is inserted, replacing the earlier copy of the page. Therefore, your manuals are always up-to-date. We make these changes to our master copies of the manuals, so that any new printings are up-to-date. If you have more than one copy of the manuals, you must reproduce the replacement pages for insertion in all copies.

Modification of software is documented in the software maintenance reports, contained in the technical summaries. These maintenance reports contain solutions to problems report in the software performance reports. The maintenance reports are divided into three forms. One form of the report is a mandatory patch. This patch contains all of the dialogue necessary to enter the change to the system. User input is underlined to distinguish it from computer output. All mandatory patches (see Appendix F) should be made, even if you are not concerned with the problem fixed by the patch. We must have all software at the same patch level to maintain proper support.

If you don't have the license for the product which is changed, ignore the modification. Entry of these mandatory patches are

discussed further in the next section.

The second form of the software maintenance report is the optional patch (see Appendix E). These patches are not fixes for software errors or bugs. Optional patches change the software in a manner which is not desirable to every user. It is your decision whether or not to implement these patches.

When we release a technical summary, we make all mandatory patches to our master distribution pack. No optional patches are made. Therefore, any packs which are sent out from QUODATA are always up to the patch level of the most recent summary. When a distribution pack is sent to a new ETOS customer, we include all technical summaries which have been distributed since the last release of ETOS. The reason for this inclusion is that we want every user to have all of the information released by QUODATA. Mandatory patches do not have to be performed on this new distribution pack. The distribution pack contains the release of ETOS (V5B) followed by an asterisk (*) and a number, representing the last technical summary patches which have been implemented. The number corresponds to the overall technical summary number which is indicated in the first sentence of the summary.

It is necessary that software be kept standard for proper support. Any user modifications to the system might cause an invalidation of the warranty. When we receive notification of changes to ETOS, we will respond as to what effect these changes will have on your system. To insure that a user requesting support has implemented all required patches, every program under ETOS has been assigned a release number. Every mandatory patch changes the release number of the particular product. The general form of the release number is X.YYZ, where X is the version number, YYY is the edit number and Z is the patch level. After a release of ETOS, Z is the only number that changes. Z is initialized at Ø and is incremented by 1 in each mandatory patch.

The overall version number for ETOS is V5B. The specification release number for ETOS.SV is Ø5.2ØZ, where Z is the patch level.

8.12 MEDIA MANAGEMENT

The technical summaries contain patches which all users are required to enter in their system. It is critical that these patches be entered exactly as written. This poses an important problem, particularly at sites where the system manager is not an experienced computer person familiar with the general rules of properly managing computer systems.

The proper operation of an ETOS system requires at least four disk packs:

Pack 1 - Original ETOS distribution pack from QUODATA. This pack should never be modified in any way. It is rarely used, and it should always be write-locked when it is used.

Pack 2 - SMR pack. This pack contains all SMR (Software Maintenance Report) corrections.

When you install ETOS, create this pack with RKCOPY or SICOPY by copying an image of the ETOS distribution pack (pack 1) onto a scratch pack, which becomes the SMR pack. Whenever you receive a technical summary, enter the patches (alias SMRs) onto this pack. Thus, the SMR pack is a completely up-to-date version of the ETOS distribution pack. It includes any modifications to configure the pack for the user's installation, such as CONFIG options. Once the SMRs have been correctly entered onto the SMR pack, the modified files and system heads may be copied onto the Production pack.

Pack 3 - Production pack. This is the pack that is used as the ETOS system device in the daily operations of the installation. It is fully configured for the specified needs of the particular site. Some sites may have more than one production pack.

Pack 4 - Production Daily Backup. This pack is an image of the Production pack. It is used to restore data lost through either human or system error. It is always good form to have a backup pack, and in some cases it is vital. Once a day, RKCOPY or SICOPY should be used to

copy the Production pack to the Production Daily Backup Pack.

Pack 5 - Production Weekly Backup. Similar to the Daily Backup, but this pack provides access to older information. It is valuable because one does not always immediately realize that he has deleted an important file or that the system is corrupted, so the daily backup may be made before the data has been recovered. A Weekly Backup Pack provides additional security against data loss.

Most industrial installations should maintain both a Daily Backup and a Weekly Backup. The time it takes to do so is minimal. However, some noncritical users, particularly schools, may not wish to tie up two disk packs for backup. We strongly advise that at least one form of backup be performed. The decision of whether to perform a Daily or a Weekly backup depends on the usage of the system and the type of data that the system manager is most interested in preserving.

8.12.1 Entering SMRs

As noted before, it is critical that the system manager correctly enter his SMRs, and that he keep track of system modules to avoid patching corrupt or improperly modified files. In order to correctly enter a SMR, all of the preceding required SMRs for the particular component must have been correctly entered. For example, if you are installing the SMR for CONFIG.SV sequence #, then CONFIG.SV sequence 1-2 must have already been installed. Moreover, no other modifications should have been made to the module, such as an early patch received over the telephone. If you maintain the four packs described earlier, and promptly enter SMRs, there should be no problem.

After each SMR has been installed, you should double-check your output to insure that no errors or irregularities occurred. All files (e.g., FILE BCOMP.SV), access modes (e.g., SET MODE SAVE), addresses, old contents (typed out), and new contents (entered) should match, with the following exceptions. The text "xxxx", or generally anything in lower case text in the SMR indicates data

that is not expected to match the user's actual output. The output of this command contains block numbers for the location of the program, which will probably be different at each site. Also, certain documentation conventions are used which will not themselves appear in the user's output. For example, all data which the user enteres is underlined in the SMR report. When underlined text is the last item on the line, it should be terminated with a carriage return, unless otherwise indicated. The text "<CR>" and "<LF>" is used to explicitly denote what terminating character should be used. "<CR>" stands for the carriage return key, and "<LF>" stands for the linefeed key.

It is a good idea to back up your SMR pack (pack 2) onto a scratch pack before entering SMRs. This way, if you do make any mistakes, or if irregularities turn up, it is easier to recover.

All functions of the system manager should be performed on a hard copy terminal. You should save your hard-copy terminal output from every SMR entering session in a permanent file. This hard copy should be double-checked for errors at the end of the session, as described above. It can be checked later if problems arise, and it may be necessary for QUODATA personnel to examine this material.

Obviously, you should always use a hard-copy terminal rather than a scope while entering SMRs. If this is not possible, there is an alternate method of obtaining hard copy which you should use. Most patches are entered with FUTIL. FUTIL has an option to allow you to store your entire terminal dialog in an OS/8 file. This method is not as good as using a hard-copy terminal, but because the non-FUTIL parts of the dialog are not saved, but it is far better than nothing. You direct FUTIL to save its terminal dialog on the line printer with the commands of Figure 8-16.

Figure 8-16
Output of FUTIL Dialogue to Line Printer

```
.R FUTIL                (this command is part of the SMR)

!VASSIGN LPT            (insert these 3 commands into SMR flow
SET DDEV LPT             immediately after entering FUTIL to
SET DMODE ALL            save the dialog on the LPT)
(continue with SMR)
    .
    .
    .
WRITE                   (part of SMR)
CLOSE                   (insert this just before EXIT - closes
                         LPT file)
EXIT                    (part of SMR)
```

The line printer output should be used to double-check the SMR entry, and it should also be saved in the permanent file discussed above.

There is a further technique that may be of interest to users who have the OS/8 EXTENSION KIT. Both FUTIL and CONFIG are compatible with OS/8 BATCH. Moreover, in ETOS V5B, OS/8 BATCH has been modified so that it will handle SCALE commands. Using BATCH streams has advantages in that it gives you better control over what gets done, and mkes it far simpler to re-enter SMRs. If OS/8 BATCH is used, listings of the BATCH streams and the BATCH log should be kept in your permanent file of terminal output. The example of Figure 8-17 should give you some idea of how to prepare the BATCH streams.

Figure 8-17
Sample FUTIL Input Batch

```
$JOB ENTERS ETOS V5A SMR FOR OS/8 BASIC, SEQ 1
!PRIV 4040
!WE 0
!PRIV 40
.R FUTIL
FILE BASIC.FF
SET MODE SAVE
13402/7775                (Note that linefeeds may be
        4550               entered into the BATCH stream
           4477            just as you would type them on a
              4477         terminal)
                 3230
                    3230
                       1206

WRITE
   .
   .
   .
1322/2201
WRITE
EXIT
!PRIV 4040
!WL 0
!PRIV 40
$END
```

8.12.2 Maintaining The Production Pack

All of the system manager's functions should be performed on a hard-copy terminal. The terminal output should be saved in a permanent file that documents all modifications related to the system management. Such modifications include entering SMRs onto the SMR pack, copying SMR mods onto the Production pack, initial building and configuration of the Production pack, and all modifications to the account structure. Such data may be crucial in the diagnosis of any problems that occur. If it is impossible to use a hard-coy terminal, you should make an effort to obtain hard-copy documentation through the line printer. The ACCNT program's LIST command, FUTIL's SET DDEV LPT and SET DMODE ALL commands (described above), and the OCOMP program (to document

changes made) can be used for this purpose.

After entering an SMR onto the SMR pack, you can copy the modified modules to the production pack. OS/8 and COS files can be copied with OS/8's COPY command. OS/8 system heads can be copied using OS/8 PIP's /Y option. COS system heads can be copied under OS/8 or COS with the program COSCOP.SV.

APPENDIX A
PRIVILEGED SCALE COMMANDS

There are restrictions as to when certain commands may be issued. The State column illustrates these restrictions. The code used for this column is

L - Terminal must be logged in
H - Virtual machine must be halted

Table A-1
PRIVILEGED SCALE Commands

Command	Description	State	Manual Section
BRO[ADCAST] message	broadcast specified message.	LH	6.2.3
DI[SMOUNT]DKn:	logically release disk "n", on which an ETOS file-structured device is mounted.	LH	6.2.6
EC[ROF]n	print output from detached console "n".	LH	6.2.4
FO[RCE]n;command	force a command into console "n's" input buffer.	LH	6.2.4
MO[UNT]DKn:password, offset	logically mount a public pack or system pack in a non-system drive.	LH	6.2.6
PE[EK] loc[n]	examine words in physical memory, starting at location "loc".	LH	6.2.2
PO[KE]loc v1[v2[...]]	deposit V1,V2,... into physical memory, starting at location "loc".	LH	6.2.2
PR[IV] [n]	privilege job, set priority queue number and quantum.	L	6.2.1
SH[UTUP]	shuts down time sharing and returns to stand-alone OS/8.	L	6.2.7

Command	Description	State	Manual Section
WE n	Write Enable channel n	L	6.2.5
WL n	Write Lock channel n	L	6.2.5

APPENDIX B

SCALE ERROR MESSAGES

Table B-1
SCALE Error Messages

Error Message	Meaning
?ALREADY LOGGED IN	Attempt to issue LOGIN when not necessary.
?BAD CHANNEL FUNCTION	Incorrect use of a channel by a user program.
?BAD NUMBER	Too many digits were entered, no number was entered where one was required, or non octal digits were entered in an octal number.
?BAD QUEUE DEVICE	There is no spooler running that is initialized for the logical spooling device that was specified in the QUEUE command.
?CANNOT DETACH	There are no dummy consoles for the job.
?CHANNEL IN USE	LOOKUP or ENTER attempted on a channel number that is in use.
?CHANNEL NOT OPEN	The specified channel can not be used because it is closed.
?CHANNEL OPEN ON DEVICE	The device cannot be DISMOUNTed because there is at least one job with a file open on the device. Every channel of every job with a file open on the device must be CLOSEd before the device can be DISMOUNTed.
?DEVICE OWNED BY JOB n CONSOLE m	JOB n has to release this device before it can be assigned.
?FILE SPECIFICATION ERROR	An illegal (bad syntax) ETOS file specification was found.

Error Message	Meaning
?ILLEGAL IOT	An illegal IOT (possibly random data) was executed by the user program, or a nonprivileged user attempted to run a privileged program.
?JMP/JMS TO NON EXISTENT FIELD	The user program executed a CIF to a non-existent field then attempted to execute a JMP or JMS. The PC contains the address of the JMP/JMS and bits 6-8 of the Flag word contains the Instruction Field the program tried to jump to. There is no indication of what field the JMP/JMS came from. If the user wishes to locate the JMP/JMS error, all existing virtual fields must be examined at the location contained in the PC for a JMP or JMS.
?JOBMAX EXCEEDED TRY LATER	The user cannot LOGIN or execute any commands that require the use of a temporary job because the maximum number of jobs that the system can handle are already running. Someone must LOGOUT before any new jobs can be initiated.
?LOGIN PLEASE	The terminal is currently logged out.
n? FUNCTION FAILURE	A channel I/O command had an error. See 3.13 for an explanation of the error code n.
?NO SWAP TRACKS AVAILABLE	The system requires 8K of virtual memory to run LOGIN. This space was not available. Other users must release it before this user can LOG IN.
?NON EXISTENT FIELD	The virtual machine does not have that memory field available for use. Re-issue the CORE command to allocate more memory.
?NOT ENOUGH SWAP TRACKS	There was not enough free virtual memory to accommodate the desired expansion. Another

Error Message	Meaning
	user must release some virtual memory.
?NOT PRIVILEGED	The command issued is not allowed unless the account is privileged.
?PLEASE TYPE <^V>S<CR>FIRST	The user issued one of the commands mentioned in 3.15 while the VM was running. The request is ignored.
?SYSTEM SWAP ERROR	Disk error during swap.
?SYSTEM R/W ERROR	System device error. Retry.
?UNKNOWN COMMAND	SCALE did not recognize the command keyword. This error can be baffling when the user has embedded a nonprinting character such as ^Q or ^O in the command.
?UNKNOWN DEVICE	The given device name is illegal or unknown.
?%VM HLT	A HLT instruction was executed, stopping the virtual machine.
?WAIT FOR I/O	The Monitor cannot execute the command at this time.
?WASN'T ASSIGNED	The user attempted to DEASSIGN a device that was not assigned to his job.
?WHAT? WHERE?	(1) An invalid command was entered; or (2) a required argument for the DEPOSIT or EXAMINE command was not entered (address or data for DEPOSIT, address for EXAMINE); or (3) a nonexistant memory field was referenced by some command; or (4) the programmed SYSCAL function CORE deallocated the VM's instruction or data field (see 6.7.13).

APPENDIX C
GLOSSARY

Account	Entry in the master file directory, consisting of an account number, password and protection code. This entry controls access to ETOS.
Account Number	Identification number made up of two octal numbers (project number and programmer number), which uniquely define a user's storage area.
Accumulator (AC)	A 12-bit register in which arithmetic and logic operations are performed; also an input/output register. Information transfers between core memory and a slow input/output device pass through the AC.
Bit	An abbreviation of binary digit; a unit of data; a unit of data in binary notation.
Bootstrap	A technique for loading the first few instructions of a routine into storage, then using these instructions to bring in the rest of the routine. This usually involves either the entering of a few instructions manually or the use of a special switch on the computer console. The SCALE command BOOT bootstraps an ETOS virtual machine.
Break character	A character which activates the user's virtual machine.
Break mask	A 12-bit word which defines the characters which are break characters.
Buffer	An area used to temporarily hold information being transmitted between external and internal core storage devices or input/output devices and internal high-speed storage. A buffer is

	often a special register or registers, or a designated area of internal storage.
Central processing unit	The hardware unit of a computing system that contains the circuits that control and perform the execution of instructions.
Channel	A path along which data can be sent; e.g., data channel, output channel. Each job under ETOS has eight channels associated with it, which are used to access ETOS files.
Compute-bound	A program using all the computational resources of the central processor. The program is running and not waiting for I/O.
Console	A section of the computer which may be used to control the machine, correct errors, determine the status of registers and counters, determine the contents of storage, revise the contents of storage, and set jump and stop switches. Also called the front panel or operator's console. Under ETOS, each terminal acts as a virtual console. See also "Dummy console".
Core memory or Core	A storage device utilizing matrix arrays of ferrite cores which are most often used as the computer's working memory.
Core resident	Used to describe a part of the monitor which resides in core memory at all times.
Device	A mechanical contrivance or appliance, such as the PT8E paper tape reader/punch,. RKØ5 disk, DK8E-A clock, etc.

Device handler	A subroutine for controlling input/output on channels and peripheral equipment.
Directory	An area containing the names and location of user files.
DSK	Each virtual OS/8 has its own DSK.
Dummy console	A console controlling a job that does not have a physical terminal associated with it. The job can be accessed via SCALE's ATTACH, ECROF, FORCE and TALK commands.
ETOS	Extended Timeshared Operating System.
Executive	See Operating system.
Extended Arithmetic Element (EAE)	An extended hardware option which performs addition, subtraction, multiplication, and division.
External storage	The storage of information on a device which is not an integral part of a computer, but is a form prescribed for use by the computer.
Field	A 4096-word block. PDP-8 memory is divided into fields numbered from 0 to 7 depending on the amount of memory installed.
Flag word	A 12-bit word which is the processor flag word containing the status of the virtual machine.
Internal storage	The storage of information on a device which is an integral part of a computer. Most internal storage devices are magnetic cores or magnetic drums.

Interrupt	A signal which when activated causes the hardware to transfer program control to some specific location in internal storage thus breaking the normal flow of the program being executed. After the interrupt has been processed, program control is again returned to the interrupted program.
I/O wait	The state of a user program while it is waiting for input/output to be completed before resuming execution of the program.
Job	A virtual machine from the monitor's standpoint. Allowed to make requests on the system for computation, input/output, and memory. Under ETOS, jobs are assigned sequentially.
Keyboard echo	The ETOS monitor echoes the characters entered on the keyboard.
Keyboard monitor	Provides communication between the user and the OS/8 executive routines by accepting commands from the console terminal. The commands enable the user to create logical names for devices, run system and user programs, save programs, and call ODT.
Location	An address in memory. On the PDP8, each location is a twelve-bit word.
Login	Assign and initialize a virtual machine and bootstrap OS/8.
Logout	Release the virtual machine from the system.
Mapping	Conversion of a virtual request to a physical request. All I/O is mapped under ETOS.
Multiplier Quotient	

(MQ)	A 12-bit bi-directional shift register that acts as an extension of the AC during EAE operations.
Normal mode	When the terminal is communicating with the user program, as opposed to being in SCALE mode.
Operating system	An integrated collection of programs that automatically permits continuous job processing by a digital computer. It is often called a monitor, supervisor, or executive.
Peripherals	Various units or machines that are used in combination or conjunction with the computer, but are not part of the computer itself, such as line printer, typewriters, mass storage devices, etc.
Physical	Pertaining to the actual machine. This term contrasts the term "virtual".
Program Counter (PC)	A 12-bit register that is used to control the program sequence. The PC contains the address of the core memory location from which the next instruction is taken.
Programmer Number	Two digit octal number, used to further segment a project number.
Project Number	Two digit octal number, used to divide accounts into logical groupings, such as departments.
Quantum	The smallest unit of central processor time available to a user. For ETOS this is 100 milliseconds.
Register	A device for the temporary storage of one or more words to facilitate arithmetical, logical, or transferral operations. A term to designate a specific computer

	unit for storing a group of bits or characters.
Resident monitor	That part of the ETOS monitor which is always located in core memory and is core resident.
Run queue	A list, line or queue of jobs ready or submitted for processing, but which has not yet begun to run.
SCALE	An acronym for System Command and Login/Logout Executive.
SCALE mode	Keyboard input is interpreted by SCALE as system commands.
Stand-alone	Pertaining to a single user operating system. This term contrasts the term time-sharing.
Step Counter (SC)	A 5-bit register which is part of the EAE option. Used to record the number of steps performed and to stop many EAE instructions after the correct number of operations.
Swap Region	The region on the system high-speed mass storage device where virtual memory fields reside when they are not in physical core memory.
Swapping	Transferring copies of user core memory between main memory and some high-speed mass storage device. Without swapping, all user jobs would have been resident in main memory at once.
Switch Register (SR)	The 12 switches which provide a means of communication between operator and machine. ETOS simulates the switches with a 12-bit SR register.

SYS	Read-only storage device containing all OS/8 public files and programs.
System	An integrated structure of hardware, software, data structures, and documentation which implements a certain function or set of functions. For example, ETOS, OS/8 and COS are all systems.
System device	A device which stores the user accounts and system files.
Terminal	An input/output device designed to receive data in an environment associated with the job to be performed and capable of transmitting entries to and obtaining output from the system of which it is a part.
Time-sharing system	A computer system in which the available computer time is shared among several users.
User mode	Hardware mode which disables I/O instructions, HLT and OSR from having their usual effect of stopping the physical machine. This protects user programs from each other.
Virtual	Generally pertains to an entity that ETOS simulates for the user, so that many jobs may share the corresponding physical resource.
Virtual Core, or Virtual Memory	Simulated core memory used by the virtual machine. The use of swapping makes it possible to have much more virtual memory on the system than physical memory. Each virtual memory field must have a corresponding swap region.
Virtual machine (VM)	The PDP-8 simulated by ETOS.

Word	A set of characters that occupies one storage location (12 bits for the PDP-8) and is treated by the computer circuits as a unit and transported as such.

APPENDIX D
TECHNICAL SUMMARY LETTERS

D.1 INTRODUCTION

Technical summaries are periodic mailings from QUODATA which allow us to support a large number of users in the most effective manner. The concept of these summaries is contained in Section 8.11. In the front of each technical summary, there is a letter which identifies the technical summary number and release date. This letter contains announcements of new products or new policies, clarification of existing policies, and items of general interest. The letter also acts as a forum for user communication. If you have written a program of general interest or require a particular program, we will include a description of your program or request in a technical summary letter.

It is obvious that some of the information in the technical summary letters may be obsolete in several months. Hardware prices are an example of this type of information. However, much of the information (e.g., descriptions of user programs) is of interest to users for the life of the product. Appendix D is used to contain those letters. When you receive the technical summary, the letter should be inserted at the end of this appendix. D. Page numbers are printed at the bottom of the letters to insure that they are placed in the current sequence. The first page of a letter contains a page number in the form "D-page". Subsequent pages of the same letter have page numbers in the form "D-page letter". Sample page numbers are D-3 and D-3A.

If you have received ETOS after the initial release of September, 1979, you may have several technical summary letters inserted in your Manager's Guide. These are all of the letters which have been distributed since the initial release of ETOS.

APPENDIX E
OPTIONAL PATCHES

E.1 INTRODUCTION

The ETOS system is designed for general utility by a wide array of users. Decisions are made about how different elements of the system should work for the majority of ETOS applications. Since each installation is unique, some users wish to modify aspects of ETOS. For example, ETOS is designed with many security procedures. Many commercial users are not concerned with security. Therefore, there needs to be procedures for overriding the default security procedures.

All programs are designed to be flexible. Any program which is used for a variety of applications (e.g., LOOKUP) has documented procedures for tailoring the program to an individual installation. We also distribute the source for ETOS CUSPs, device handlers and assembler information. This distribution allows modification by sophisticated users of ETOS. Even with the flexibility provided, there are changes to ETOS which some users require but are unable to make themselves due to lack of assembler knowledge or the absence of the source of the monitor.

Changes which are of interest to a number of users are handled via optional patches. Optional patches do not change program version numbers because they are not required for all users. Four optional patches are distributed with the original ETOS V5B release. Additional optional patches are distributed in technical summaries (see 8.11), which are sent periodically from QUODATA. These additional patches are inserted by you into the manual and have page numbers on them which fit properly in this appendix. Every group of optional patches also requires that you change the table of contents to reflect the additional subsections of Appendix E.

E.2 SETTING LOGIN PRIVILEGE

When you LOGIN to the system, LOGIN automatically sets your PRIV word (see 6.2.1) to 0040. This word indicates that you are currently a non-privileged (even if you are in a privilegable account) user at position 4 on the priority queue, with a quantum specified by entry 0 in the LSLICE table (see 3.13). If you are in a privilegable account, you can change your PRIV word to any value. The most common value of the PRIV word other than 0040 is 4040, which indicates that you are privileged and can write on SYS, initialize public packs and perform other privileged operations.

You can change the PRIV word which LOGIN automatically sets for all users. You might want users to have a different priority number or quantum. However, the most common setting of the PRIV word is 4040. This setting should only be used in installations where security is not an issue. All accounts in the system must be defined as privilegable (see 5.2.1) or the non-privileged accounts will allow users to be partially privileged. The setting of 4040 is often used at sites, where all users must access a floppy disk or a DECtape, since these devices require privilege.

Location 00111 in the ETOS file [0,2]LOGIN.SAV contains the default PRIV word for all users. On the virtual OS/8 system device, there exists a file called LOGIN.SV which contains a copy of the ETOS file LOGIN.SAV. Therefore, location 00111 in LOGIN.SV can be modified with FUTIL and the revised OS/8 file can be transferred to the ETOS file. This procedure insures that you have a current back-up of the ETOS LOGIN file. If you have created your own version of LOGIN from the source distributed in account [00,07], modify the source file, create an OS/8 save image file and transfer the revised OS/8 file to the ETOS file.

In either case, the OS/8 program OSETOS (see 5.12.3) is used to transfer the OS/8 version of LOGIN to the ETOS file. Figure E-1 contains a dialogue which sets the default PRIV word to 4040. You must be LOGged IN to a privileged account to follow this dialogue.

Figure E-1
Modifying LOGIN to Set Privilege

```
.^VPRIV 4040
^VWE 0
R FUTIL
FILE LOGIN
LOGIN.SV  ssss-eeee  0012 (0010)  b.111  04-AUG-79
SET MODE SAVE
111/ 0040 4040<RET>          (revised PRIV word)
WRITE
EXIT
.R OSETOS
'OSETOS' V1.032
*SYS:LOGIN.SV
FILE LOADED
.^VWL 0
```

Each time that you use the Sysgen option of ETOS (see 4.2), you must modify the ETOS file [0,2]LOGIN.SAV. However, there is no need to patch the OS/8 file LOGIN.SV each time you must modify the ETOS file. You need only to transfer the previously modified LOGIN.SV to the ETOS file via the OSETOS program. The dialogue of E-1 can be used, starting with the command R OSETOS.

As an alternative to transferring a new copy of LOGIN from OS/8 to ETOS each time a Sysgen is performed, the LOGIN image which is created by the Sysgen can be modified. Even if this change is made, the dialogue of Figure E-1 should be followed to affect the change immediately. If this is not done, the modification does not take effect until the next Sysgen is performed. Figure E-2 contains the dialogue necessary to modify the LOGIN image created by Sysgen. This dialogue must be followed under stand-alone OS/8.

Figure E-2
Modifying Sysgen to Set LOGIN Privilege

```
.R FUTIL
FILE ETOS
ETOS.SV  ssss-eeee  0165 (0117)  b.111  30-SEP-79
(A+154).111/ 0040 4040<RET>    (new PRIV word)
WRITE
EXIT
.
```

E.3 DISABLING CUMULATIVE STATISTICS FILES

When you LOGOUT of the system, the ETOS LOGIN program automatically creates or updates an ETOS file, which contains cumulative usage statistics for the account. The ETOS files are created in account [Ø,3] with the name ACCTIMX.YYY, where X is the first digit of the account number and YYY represents the last three digits of the account number. The cumulative statistics file for account [12,7Ø] is called ACCTIM1.27Ø. The OS/8 program USAGE (see 6.4.14) is utilized to print these statistics. If you are not interested in cumulative usage statistics, these ETOS statistics files are annoying. They take up one block per file, they take up space in an ACCNT listing and they are not deleted automatically when an account is deleted. Therefore, they tend to remain on the disk for long periods of time.

You can modify LOGIN, so that these account statistics files are not created automatically. The change also changes LOGIN so that if an account statistics file exists for the account, it is deleted. This procedure automatically cleans the system disk of these files in a period of normal usage. Account statistics files, which correspond to accounts which have previously been deleted, must be deleted manually via the ACCNT program (see 5.6).

On the virtual OS/8 system device, there exists a file called LOGIN.SV, which contains a copy of the ETOS file [Ø,2]LOGIN.SAV. Therefore, LOGIN.SV can be modified with FUTIL and the revised OS/8 file can be transferred to the ETOS file. This procedure insures that you have a current backup of the ETOS LOGIN file. If you have created your own version of LOGIN from the source distributed in account [ØØ,Ø7], modify the source file, create an OS/8 save file and transfer the revised OS/8 file to the ETOS file. In either case, the OS/8 program OSETOS (see 5.12.3) is used to transfer the OS/8 version of LOGIN to the ETOS file.

Figure E-3 contains the dialogue which disables the creation of the account statistics files. If you ever wish to reenable use of cumulative statistics, restore the original contents, which are printed by FUTIL in Figure E-3.

Figure E-3
Disabling Cumulative Statistics Files

```
.^VPRIV 4040
^VWE 0
R FUTIL
FILE LOGIN
LOGIN.SV  ssss-eeee  0013 (0011)  b.111  04-AUG-79
SET MODE SAVE
1407/  7450  7650<RET>
1411/  1375  5774<RET>
1424/  1370  6100<LF>
0000.01425\ 6200 7300<LF>
0000.01426\ 4443 5774<RET>
1571/ 0001 1000<RET>
1577/ 0400 0500<RET>
WRITE
EXIT
.R OSETOS
'OSETOS' V1.032

*SYS:LOGIN.SV
FILE LOADED

.^VWL 0
```

Each time that you use the Sysgen option of ETOS (see 4.2), you must modify the ETOS file [0,2]LOGIN.SAV. However, there is no need to patch the OS/8 file LOGIN.SV each time you must modify the ETOS file. You need only to transfer the previously modified LOGIN.SV to the ETOS file via the OSETOS program. The dialogue of E-3 can be used, starting with the command R OSETOS.

As an alternative to transferring a new copy of LOGIN from OS/8 to ETOS each time a Sysgen is performed, the LOGIN image which is created by the Sysgen can be modified. Even if this change is made, the dialogue of Figure E-3 should be followed to affect the change immediately. If this is not done, the modification does

not take effect until the next Sysgen is performed. Figure E-4 contains the dialogue necessary to modify the LOGIN image created by Sysgen. This dialogue must be followed under stand-alone OS/8.

Figure E-4
Modifying Sysgen to Disable Cumulative Statistics Files

```
.R FUTIL
FILE ETOS
ETOS.SV  ssss-eeee  0165 (0117)  b.111  30-SEP-79
(A + 157).7/ 7450 7650<RET>
11/ 1375 5774<RET>
24/ 1370 6100<LF>
bbbb.00025\ 6200 7300<LF>
bbbb.00026\ 4443 5774<RET>
171/ 0001 1000<RET>
177/ 0400 0500<RET>
WRITE
EXIT
.
```

E.4 MAKING THE TALK COMMAND PRIVILEGED

Users at different terminals can communicate with each other via the TALK SCALE command (see 3.9, System User's Guide). This command is particularly useful if you have users at remote terminals who need paper put in the line printer, a DECtape mounted, or some other operator function performed. However, on a system with hostile users, this command can be abused. Insulting messages may be sent, sample runs for documentation may be interrupted and novice users may be discouraged from the use of the computer by critical messages.

The solution to the problem of users abusing the TALK command is to make the TALK command privileged. After this change is made, if a non-privileged user attempts to use the TALK command, the message "?NOT PRIVILEGED" is printed. Figure E-5 contains the dialogue which is followed in order to make the TALK command privileged. This dialogue must be followed under stand-alone

OS/8.

Figure E-5
Making the TALK Command Privileged

```
.R CONFIG
'CONFIG V3.012
*KMON,05703=7333
KMON(05703) 7000 7333
*KMON,05704=4523
KMON(05704) 7000 4523
*KMON,05705=0005
KMON(05705) 7000 0005
*KMON,05706=7650
KMON(05706) 7000 7650
*KMON,05707=5767
KMON(05707) 7000 5767
*KMON,05767=6650
KMON(05767) 7000 6650
*EXIT
.
```

E.5 DISABLING CHANIO COMMANDS

There are nine non-privileged SCALE commands which are used to manipulate ETOS channels. These commands are known as CHANIO commands (see 7.2, System User's Guide). These commands are used to manipulate (e.g., create, delete, change the length of) ETOS files. On a system with hostile users, these commands can be abused. Variable length ETOS files, such as OS/8 or COS storage areas may be EXTENDed or REDUCEd. All of the available space on the disk could be wasted in this manner. Another common problem is the deletion of the OS/8 scratch blocks file on channel 1.

The solution to these problems of users abusing the CHANIO command is to disable these commands. After this change is made, if a non-privileged user attempts to use a CHANIO command, the message "?UNKNOWN COMMAND" is printed. If the SCALE commands are disabled, the program CHANIO IOTs (see 7.4, System User's Guide) may still be utilized by non-privileged users. Therefore, the LOOKUP program (see 5.14) may be run by non-privileged users. This allows you to disable direct use of all CHANIO operations except LOOKUP. LOOKUP is often used to facilitate access of an auxiliary storage area or a common program library. If you wish to disable LOOKUP also, delete the LOOKUP program from the virtual OS/8 system area.

Figure E-6 contains the dialogue which is followed in order to make the CHANIO commands privileged. This dialogue must be followed under stand-alone OS/8.

Figure E-6
Disabling CHANIO Commands

```
.R CONFIG
'CONFIG' V3.012
*KMON,02330=000
KMON(02330) 3531 0000
*KMON,023331=5355
KMON (02331) 1513 5355
*EXIT
.
```

APPENDIX F
MANDATORY PATCHES

F.1 INTRODUCTION

In the periodic technical summaries (see 8.11), mandatory patches are distributed on yellow software maintenance report forms. These patches are solutions to software errors or bugs in some element of the ETOS system. The patch contains all of the dialogue necessary to implement the patch into the system. Section 8.12 discusses the entry of the patches and proper backup procedures for the modifications.

Unlike optional patches, mandatory patches must be made by all users who are running the software module being patched. Obviously, if you don't have a COBOL license, you can't make any mandatory patches to the COBOL product. All mandatory patches should be made, even if you are not concerned with the problem fixed by the patch. A subsequent patch may depend on a previous patch having been implemented. Every mandatory patch changes the version number of the program being patched. The patch contains the current version number of the program being patched so that you can insure that you have a current copy of the program. Each patch also has a sequence number which is the relative number of the patch corresponding to a particular program. The first patch made to a program has a sequence number of 0001.

When you receive a technical summary, insert the mandatory patches in this Appendix in the order in which they are distributed. The patches are distributed alphabetically by product name. An index for the mandatory patches must be inserted at the end of Section F.3, prior to the actual patches. The page numbers for the indexes are in the form "F3-letter", where the letter indicates the sequential position of the index. The page numbers for the patches are in the form "F- page number-

letter", where the page number increments by one for each patch and the letter indicates the position of multiple pages of a patch. The index provides you with the starting page number of each patch distributed in the technical summary corresponding to the index.

F.2 MANDATORY PATCH INDEX

This section contains one index page for each technical summary released by QUODATA after your original distribution of ETOS. The index contains a list, alphabetically by product name, of all mandatory patches distributed in the particular technical summary. Since the patches are inserted at the end of Appendix F, the index also contains the starting page number of each patch. Your distribution manual contains no indexes, because all mandatory patches released to date have been installed in the distributed copy of ETOS. Therefore, the previously implemented mandatory patches, and the indexes to these patches are of no interest. If there had been four technical summaries distributed prior to the distribution of your system, the first index received from QUODATA has a page number of F3-E. All mandatory patches distributed after your distribution of ETOS must be inserted at the end of Section F.3 in this appendix.

When we release a technical summary, we make all mandatory patches to our master distribution pack. No optional patches are made. Therefore, any packs which are sent out from QUODATA are always up to the patch level of the most recent summary. Unlike technical summary letters (see Appendix D) and optional patches (see Appendix E), the Manager's Guide does not contain all mandatory patches distributed since the release of the product. Since all mandatory patches have been made, there is no reason to distribute them to you. Therefore, Appendix F never contains patches upon distribution. Patches are inserted only from subsequent technical summaries. Page numbers for the indexes and the patches you receive might not start at F-3-A and F-5. If one technical summary and four mandatory patches had been released prior to your distribution of the system, your first index contains a page number F-3-B. Your first patch starts with page number of F-9. The mandatory patches contained in the first technical summary are of no interest and if included, would only

cause confusion.

F.3 SOFTWARE MAINTENANCE REPORTS

Mandatory patches are distributed in the form of software maintenance reports. Insert all software maintenance reports in sequence after this page.

INDEX

USER COMMENT FORM

MANUAL:

Please use the space below for any comments you may have concerning this manual. Our goal is to produce documentation that is clear, complete and useful, and you, the reader, are the final judge.

We are especially interested in any thoughts you may have concerning this manual's clarity, completeness, accuracy, structure and overall effectiveness.

Please use this form only for comments about this manual.

NAME: _____

OCCUPATION: _____

ADDRESS: _____

_____