CHAPTER 5
ETOS ACCOUNTS

5.1 INTRODUCTION

The previous chapters discussed the installation, configuration and operation of a one-user ETOS system. This chapter contains information which will allow you, as system manager, to enable ETOS to handle multiple users.

At this point, only the console terminal may be LOGged IN. To do so, you must specify an account number (Ø,3) and the password. Refer to 2.2.2 of the ETOS Users Guide for LOGIN procedures. The procedure for the console is contained in Figure 5-1.

Figure 5-1
LOGIN of Console Terminal

```
.R ETOS
ETOS V5
OPTION?T
[carriage return to bring LOGIN]
?LOGIN PLEASE
!LOGIN
ETOS V5B AT 10:03:12 A.M. ON TUE 01-JAN-79
ACCOUNT? 0,3
PASSWORD? OPERATOR    (not echoed)
JOB  3 LOGGED IN ON CONSOLE   0

?CANNOT FIND OS8DISK.DSK
.
```

This chapter describes the user account structure. The ETOS files, and the ACCNT program which is used to manipulate accounts and ETOS files.

5.2 ACCOUNT STRUCTURE

Under stand-alone OS/8 or COS operations, one user has access to the entire RKØ5 or System Industries disk. Under ETOS, the system manager assigns a section of the disk to each user. This section is known as an ETOS account. In addition, the system manager may create ETOS files associated with the ETOS account. These could be used as OS/8 storage areas, COS storage areas, and/or other specialized ETOS files.

The user appears to have his own PDP-8 with a prescribed disk area. The only apparent major difference between processing under stand-alone OS/8 or COS and ETOS is the size of the disk.

When you enter the system via the LOGIN process (see 2.2.2, System User's Guide), you have access to a common area of OS/8 programs (e.g., DIRECT, PIP). If OS/8 storage has been set up for this account, you also have access to this storage area. If you intend to use COS, you must have a COS storage area associated with your account.

Two users should not be LOGged INto the same account. If another user LOGINs to your account which you are already LOGged INto, a warning message is printed and your OS/8 account storage area (if it exists) is write protected. If you want to create OS/8 files or run COS, you should LOGOUT of the account and LOGIN to a different account. The reason that you can't have two users creating files in the same OS/8 disk area is that the OS/8 directory structure prohibits the use of more than one temporary file. Two users can access files in the disk area if they are not changing the directory. The reason that you can't have two users running COS in the same account is that the scratch blocks would become jumbled.

5.2.1 Account Attributes

Each ETOS account on the system has four major attributes.

Accounts have an account number which consists of a project number and programmer number. The project number is a general grouping and the programmer number is a specific individual. For example, all members of a certain department might have the same project number. Each individual in that department has a different programmer number. The project and programmer numbers are octal numbers in the range ØØ to 77. A sample account number is [Ø7,52] (note: account numbers are represented with the square brackets). In this sample account number, Ø7 is the project number and 52 is the programmer number.

The second attribute for an account is a password. This password must be specified when you LOGIN to the system. It is not echoed. The password gives each account security which is maintained by limiting the users right to know the list of operating passwords. The password consists of one to eight alphanumeric characters. A sample password is OPERATOR.

The third attribute for an account is its protection code. This protection code determines whether or not a user who LOGs IN to the account is privilegable. Privileged users can delete system programs, SHUT the system down and access anyone's account. For this reason, the number of privilegable users should be kept at a minimum. Bit Ø of the protection code enables privileged status. If bit Ø is set, the account is privilegable. Even if an account is privilegable, the user must execute the PRIV command (see 6.2.1) to allow privileged processing. The protection code also controls the amount of virtual memory which is automatically assigned to the user at LOGIN time. The user can always obtain more memory via the CORE command (see 3.4, System User's Guide). The protection code allows you to automatically assign between 8K

THIS IS PRIVELEGE!

and 20K words when the user LOGs IN to the account. Bits 1 and 2 control the amount of virtual memory assigned. Since OS/8 and COS require 8K, the minimum amount of memory you can assign is 8K. Bits 1 and 2 represent the additional amount of memory automatically assigned. Bits 3 through 5 in the protection code are always set to 0. Bits 6 through 11 are always set to 1. All possible values for the protection code and their meaning are listed in Table 5-1.

Table 5-1
Account Protection Codes

Protection Code	Meaning
0077	Non privileged, 8K words
1077	Non privileged, 12K words
2077	Non privileged, 16K words
3077	Non privileged, 20K words
4077	Privileged, 8K words
5077	Privileged, 12K words
6077	Privileged, 16K words
7077	Privileged, 20K words — Tom + John

The final attribute of an ETOS account is the date of creation. Unlike the other attributes, there is no control over this attribute. It is automatically assigned the date of the account's creation.

The ETOS user account is created through the ACCNT program which records the account and its attributes in the Master File Directory (MFD). The MFD is an ETOS file under account [00,01]. Each account which is created has a constant overhead of one block for every sixteen ETOS files created in it. This space is used for the User File Directory (UFD), which contains a list of all ETOS files in a user account.

5.2.2 ETOS File Attributes

Each ETOS account created may have several ETOS files contained in it. These ETOS files may be OS/8 storage areas, COS storage areas, spooling temporary files, EDUCOMP BASIC temporary files, etc.

The list of ETOS files created for a specific user account are listed in a User File Directory (UFD). The contents of this directory can only be accessed using the ACCNT program. The UFD is an ETOS file with the user's account number, [xx,yy].

All ETOS user files can be identified by the prefix of the account number, [xx,yy], appended to the file name.

The ETOS user file structure is summarized in Figure 5-2.

Each ETOS file in an account has five attributes. It has a file name. The file name consists of one to seven alphanumeric characters for the file, a period (.), and one to three alphanumeric characters for the file extension. There are several file names which are reserved for special purposes. "OS8DISK.DSK" is used for individual OS/8 storage areas. "COSDISK.RTS" is used for individual COS storage areas. "DATAF1.LES" is used for COS data file access. Files associated with ETOS accounts are represented in the documentation with a prefix giving the account number, [XX,YY]. For example, the OS/8 storage area for user of account [17,51] is listed as [17,51]OS8DISK.DSK.

Figure 5-2
ETOS User's File Structure

Account [ØØ,Ø1] Master File Directory (MFD)	contains record of established account and their attribut - point to location of UFD
Account [XX,YY] User File Directory (UFD)	contains record of user ETOS files and points to their locations(s)

OS/8 Storage Area	COS Storage Area	Additional ETOS Files
[XX,YY]OS8DISK.DSK	[XX,YY]COSDISK.RTS [XX,YY]DATAFILES	[XX,YYBASIC.TMP [XX,YY]TEMP.LPT [XX,YY]LOGMESG.TXT

OS/8 Files	COS Files

In addition to OS/8 or COS storage areas and COS data file areas, there are also ETOS files connected to the management of the ETOS system. These have names which describe their function. For example, the file containing the swap tracks (the virtual core of all users) is called [ØØ,Ø2]SWAPTRA.CKS.

The second attribute for an ETOS file is a length. This length is the number of blocks (256 words or 512 characters) which are assigned to the file. These blocks are removed from the free

block pool and are allocated to the file. If the file is deleted at a later date, the blocks assigned to it are returned to the free block pool. The ETOS file [ØØ,Ø3]FREEBLO.CKS contained in account [ØØ,Ø3] controls the free block pool. If the ETOS file is a storage area for OS/8 or COS, the length is the number of blocks allocated for private storage of files. However, there is a constant overhead for each storage area which is not available for user storage. Each OS/8 storage area ([XX,YY]OS8DISK.DSK) has a system overhead of seven blocks, which is used for the OS/8 storage area directory. Each COS storage area ([XX,YY]COSDISK.RTS) has a system overhead of ninety-six blocks which is used for the COS operating system, SYSGEN table and directory. Therefore, if you wish to save files totaling fifty blocks in an OS/8 storage area, assign fifty-seven blocks to the area. If you wish to save files totaling fifty blocks in a COS storage area, assign one hundred and forty-six blocks to the area.

The third attribute of an ETOS file is the file protection code. The file protection code is completely separate from the account protection code. The account protection code controls privileged status and automatic assignment under ETOS.

The ETOS file protection code controls read and write access to the ETOS file. If the ETOS file is an OS/8 or COS storage area, the protection code controls access to all files in the storage area. There is no way to protect individual OS/8 or COS files within an account. Read and write protection of all files is controlled by the account number of the users, who you wish to access the ETOS file. Remember that the account number consists of a project number and programmer number. The file protection code format is specified in Table 5-2. If the bit is set to one, the protection feature is enabled.

THIS IS PROTECTION!

Table 5-2
User File Protection Code

Bit	Meaning
6	Write protect against users LOGged INto an account with a different project number.
7	Read protect against users LOGged INto an account with a different project number.
8	Write protect against users LOGged INto an account with the same project number, but a different programmer number.
9	Read protect against users LOGged INto an account with the same project number, but a different programmer number.
10	Write protect against all users, including users LOGged INto the same account.
11	Read protect against all users, including users LOGged INto the same account.

Bits 10 and 11 are not normally used, since they deny access even to the owner of the account. The two most common protection codes are 0050 and 0074. A protection code of 0050 means that all users are able to read the files in this OS/8 or COS storage area, but no users other than someone LOGged INto the account are able to write or delete files in the storage area. A protection code of 0074 means that no user who is not LOGged INto the account can read or write files in this storage area. The protection code of 0074 should be used for all account storage areas which contain sensitive data.

For the ETOS files, [XX,YY]OS8DISK.DSK and [XX,YY]COSDISK.RTS, the bits 0-5 (the high order bits) are always set to zero.

If an ETOS file will be used as an ETOS program (e.g., SYSTAT, DKCOPY), the protection code has two uses in addition to controlling read/write access. Bit 0 controls the temporary privileged status of the program. If this bit is set, a

privileged program can be run in a non-privileged user account. When the program is terminated, the user's temporary privileged status disappears. Section 6.7 contains more information about this feature.

5.2.3 Initial Account List

After a Sysgen, the ETOS pack contains only three system accounts: [00,01], [00,02] and [00,03]. Account [00,01] is the Master File Directory (MFD), which contains a list of all accounts on the system. Account [00,02] is the Library account, which contains ETOS files, which are of system-wide use. LOGIN.SAV is an example of a file, which is contained in this account. Account [00,03] is the operator's account, which contains ETOS files used for temporary storage. This is the only account out of the three system accounts which can be LOGged INto. All operator functions (e.g., initializing time and date) are performed in this account.

The three system accounts and the ETOS files in the accounts are discussed in detail in section 5.15. You do not need to know the function of each file in order to create accounts. These system accounts and files are mentioned here because they will appear on LISTings of the ETOS file structure, which are performed in the following sections.

5.3 ACCNT PROGRAM

The previous sections in this chapter have discussed the structure of accounts and ETOS files under those accounts. The ACCNT program is used to create and delete accounts, allocate ETOS files, change file lengths and perform other manipulations of accounts and ETOS files. The features of this program are discussed in this section.

5.3.1 Initiating the ACCNT Program

In order to run the ACCNT program you must be LOGged INto the operator's account ([0,3]) on the console terminal (00). This account is privilegable but you must give yourself a 4040 privilege.

Figure 5-3
Sample Initiation of ACCNT

```
!LOGIN                          (on the console terminal)
ACCOUNT? 0,3                    (operator's account)
PASSWORD? OPERATOR              (does not echo)

?OS8DISK.DSK NOT FOUND          (no storage area exists yet)
.^VPRIV 4040                    (privilege yourself)
R ACCNT                         (run the ACCNT program from SCALE)

ACCNT, VERSION 5.53

*OUTPUT<INPUT
*<RET>
```

The final <RET> in the previous dialogue signifies the default device, the console terminal, for both input and output. Sections 5.5.9 and 5.9.7 discuss the several extended choices that may be inserted into the OUTPUT<INPUT format. These choices are useful after you have created user accounts and user ETOS files. If you are running ACCNT on a newly SYSGENed disk, you must enter <RET> to the "*" of ACCNT because this disk has no user accounts and ETOS files.

After you press the carriage return key in response to the asterisk, the system responds with

```
ETOS DEVICE NAME ?:
```

The device name is of the form "DKX:", where X specifies the disk pack being accessed. For RK05 systems, X represents the drive

number (Ø,1,2 or 3); SYS: can be used in place of DKØ:. For System Industries systems, DKØ represents the fixed pack in drive Ø; DK1: represents the removable pack in port Ø; DK2: represents the fixed pack in port 1; DK3: represents the removable pack in port 1. If you are installing ETOS for the first time, enter SYS:. You cannot specify a device other than SYS: until you initialize it as an ETOS device (see 5.11). Note: after you enter SYS:, the ACCNT program equates the device to DKØ:.

After you have specified the ETOS device name, the system responds with

```
PASSWORD ?:
```

The password requested is the account password for the Master File Directory ([Ø,1]) on the specified pack. Unless you explicitly rename the password (see 5.7), the password for account [Ø,1] is "PASSWORD". After you enter the correct password, the system responds with

```
OPTION ?
```

At this point, you can specify one of the options listed in Table 5-3 and described in the following sections, in order to manipulate accounts and ETOS files.

In addition to the options which allow creation, deletion, modification and display of ETOS accounts and files, there are three commands which allow you to restart or exit the ACCNT program. If you enter RESTART in response to the OPTION?: question, the ACCNT program responds with

```
'ACCNT' V5.53

*OUTPUT<INPUT
*
```

You are now at exactly the same point as you were when you specified R ACCNT, initially.

If you enter DEVICE in response to the OPTION question, the ACCNT program responds with

```
ETOS DEVICE NAME ?:
```

You are now at exactly the same point as you were after you ran the ACCNT program initially and specified the input and output device. This command allows you to manipulate several different drives in the same run of the program.

If you enter EXIT in response to the OPTION?: question, the system prints "CHANNELS 5,6,7 CLOSED" and exits to the OS/8 monitor. ^C is equivalent to EXIT.

The ACCNT program uses ETOS channels 5,6 and 7 for its operation. At the end of the program, the channels are closed. For a discussion of ETOS channels, see Chapter 7 of the User's Guide. Before you create accounts, it is not necessary to understand the ETOS channel structure.

5.3.2 Available ACCNT Options

All valid options of ACCNT and where they are documented are contained in Table 5-3.

Table 5-3
ACCNT Options

Option	Description	Manual Section
CRE[TE]	Create account or ETOS file.	5.5
DELE[TE]	Delete account or ETOS file.	5.6
DEVI[CE]	Restart the ACCNT program at the ETOS DEVICE NAME?: question.	5.3.1
EXIT	Terminate execution of the ACCNT program.	5.3.1
EXTE[ND]	Extend the length of an ETOS file.	5.8
LIST	List an ETOS file structure.	5.4
NOPA[SSWORDS]	Inhibit printing of account passwords in listing of the ETOS file structure.	5.4
NORM[AL]	Enable normal output mode in listing of the ETOS file structure.	5.6
PASS[WORDS]	Enable printing of account passwords in listing of the ETOS file structure.	5.4
RENA[ME]	Rename account password or protection code or ETOS file name or protection code.	5.7
REST[ART]	Restart the ACCNT program at the input/output file specification.	5.3.1
SPEC[IAL]	Enable special output mode in listing of the ETOS file structure.	5.6
WRIT[E]	Force all buffered output to the specified output device.	5.9.7
ZERO	Initialize an OS/8 directory in and OS/8 account storage area.	5.5.1
^C	Same as EXIT.	5.3.1

5.4 LISTING ACCOUNTS AND ETOS FILES

The LIST option is used to list accounts on an ETOS pack.

```
OPTION ?: LIST
ACCOUNT ?:
```

After you enter LIST in response to the "OPTION?:" question, the ACCNT program prints "ACCOUNT?:". At this point, enter the account which you wish to list. All of the ETOS files (e.g., OS/8 storage areas, COS storage areas, scratch blocks) contained in the account are listed in addition to the account number, password and privilege word.

For example, to list the [0,1] Account, Master File Directory (MFD), specify [0,1] in response to the "ACCOUNT?" question. Equivalent specifications are [00,01], 0,1, 1, MFD and ACCOUNT. A list of the Master file directory on a newly Sysgened disk is contained in Figure 5-4.

Figure 5-4
List of the Master File Directory

```
OPTION?:LIST
ACCOUNT?:[0,1]

00,01 PASSWORD
    00,01   PASSWORD   <7777>   0001    1
    00,02   LIBRARY    <7777>   0001    1
    00,03   OPERATOR   <7777>   0002    2

    FREE BLOCKS:                6357   3311
OPTION?:
```

If you LIST the Master File Directory ([0,1]), a list of all accounts on the system will be printed, along with their account numbers, passwords and protection code. The general format of the Master File Directory list is contained in Figure 5-5.

Figure 5-5
MFD List Format

```
Ø0,Ø1 MFD password
    Ø0,Ø1 password <protection code> UFD size UFD size  creation
                                     (octal)  (decimal) date
    Ø0,Ø2 password <protection code> UFD size UFD size  creation
                                     (octal)  (decimal) date
    Ø0,Ø3 password <protection code> UFD size UFD size  creation
                                     (octal)  (decimal) date
    .
    .
    .

                          Number of                Number of
    FREE BLOCKS:     free blocks (octal)      free blocks (decimal)
```

In addition to the three user-specified attributes (account number, password and protection code), ACCNT also displays the UFD size and the creation date of the account. The UFD size is the size of the user file directory for the account which contains a list of all ETOS files in the account. Each file entry takes up only 16 decimal words (see 8.5) so that the size of the UFD is normally one or two blocks. There is no creation date for the accounts created by Sysgen, which are listed in Table 5-4.

Table 5-4
SYSGEN Accounts

```
[Ø0,Ø1] PASSWORD     (Master File Directory)
[Ø0,Ø2] LIBRARY
[Ø0,Ø3] OPERATOR
```

The LIST format for any account specification other than [Ø,1], is contained in Figure 5-6.

Figure 5-6
User Account List Format

```
XX,YY account password
     filename <protection code> file length file length  creation
                                (octal)      (decimal)   date
     filename <protection code> file length file length  creation
                                (octal)      (decimal)   date
     .
     .
     .

                          Number of                  Number of
    FREE BLOCKS:     free blocks (octal)        free blocks (decimal)
```

The account number and the password of the account, which you specify, are listed. Under the account number and password, all ETOS files in the account are listed, along with their file attributes. In addition to the user-specified attributes, the file creation date is displayed. After the file length in octal and in decimal an asterisk (*) is displayed if the file is a fixed length file. A list of the LIBRARY account ([0,2]) in a newly Sysgened disk is contained in Figure 5-7.

Figure 5-7
List of the Library Account

```
      OPTION?: LIST
      ACCOUNT?: [0,2]

00,02 LIBRARY
      FREEBLO.CKS     <7777>          0040*       32*
      SWAPTRA.CKS     <7777>          0600*      384*
      OS8      .OS8   <0052>          1200*      640*
      OS8      .RTS   <1152>          3300      1728
      LOGIN    .SAV   <5152>          0011*        9
      IMAGE    .RK5   <0077>         14540*     6496*
      IMAGE    .RKA   <0077>          6260*     3248*
      IMAGE    .RKB   <0077>          6260*     3248*
FREE BLOCKS:                          6357      3311
OPTION?:
```

The IMAGE.RK? files are dummy files which take up no space on the disk. Their purpose is explained in 5.15.2. The previous list

was valid for an RKØ5 system. On a System Industries disk, there are five image files which are listed in Table 5-5.

Table 5-5
System Industries Image Files

ETOS Filename	Protection Code	Length (Octal)	Length (decimal)
IMAGE.RK5	<0077>	31400	13056
IMAGE.RKA	<0077>	6300*	3264
IMAGE.RKB	<0077>	6300*	3264
IMAGE.RKC	<0077>	6300*	3264
IMAGE.RKD	<0077>	6300*	3264

In all of the sample listings given in this section, the RKØ5 image files are used. The image files and the number of free blocks are the only difference between a newly Sysgened RKØ5 system and a Systems Industries system.

If you wish to list all of the accounts on the system, enter Ø or ALL in response to the "ACCOUNT?:" question. A listing of all accounts on a newly SYSGENed disk is contained in Figure 5-8.

Figure 5-8
Listing of All Accounts

```
ACCOUNT?:0

00,01 PASSWORD
   00,01 PASSWORD  <7777>          0001              1
   00,02 LIBRARY   <7777>          0001              1
   00,03 OPERATOR  <7777>          0002              2

00,02 LIBRARY
      FREEBLO.CKS  <7777>          0040*            32*
      SWAPTRA.CKS  <7777>          0600*           384*
      OS8    .OS8  <0052>          1200*           640*
      OS8    .RTS  <1152>          3300           1728
      LOGIN  .SAV  <5152>          0011*             9
      IMAGE  .RK5  <0077>          14540*         6496*
      IMAGE  .RKA  <0077>          6260*          3248*
      IMAGE  .RKB  <0077>          6260*          3248*

00,03 OPERATOR
      JOB03  .SBK  <0025>          0030*            24*
      JOB04  .SBK  <0025>          0030*            24*
      JOB05  .SBK  <0025>          0030*            24*
      JOB06  .SBK  <0025>          0030*            24*
      JOB07  .SBK  <0025>          0030*            24*
      JOB10  .SBK  <0025>          0030*            24*
      JOB11  .SBK  <0025>          0030*            24*
      JOB12  .SBK  <0025>          0030*            24*
      JOB13  .SBK  <0025>          0030*            24*
      JOB14  .SBK  <0025>          0030*            24*
      JOB15  .SBK  <0025>          0030*            24*
      JOB16  .SBK  <0025>          0030*            24*
      JOB17  .SBK  <0025>          0030*            24*
      JOB20  .SBK  <0025>          0030*            24*
      JOB21  .SBK  <0025>          0030*            24*
      JOB22  .SBK  <0025>          0030*            24*

FREE BLOCKS:                       6357             3311
OPTION?:                          octal            decimal
```

In addition to the options of listing one account or all accounts, the ACCNT program has the capability of listing groups of accounts. There are two syntaxes which perform equivalent tasks. When the "ACCOUNT?" question is asked you can enter

[WW,XX] TO [YY,ZZ]
or
[WW,XX] - [YY,ZZ]

ACCNT will list all accounts which have account numbers greater than or equal to [WW,XX] and less than or equal to [YY,ZZ]. Sample specifications are:

```
[0,1] To [0,2]
MFD - [0,5]
[1,5] - [7,10]
ACCOUNT TO [1,3]
```

If you LIST from [0,1] to [77,77], the affect is the same as if you LIST all accounts. A sample use of LISTing multiple accounts is contained in Figure 5-9.

Figure 5-9
Listing Multiple Accounts

```
OPTION?: LIST
ACCOUNT?:[0,1] TO [0,2]

00,01 PASSWORD
   00,01 PASSWORD  <7777>        0001              1
   00,02 LIBRARY   <7777>        0001              1
   00,03 OPERATOR  <7777>        0002              2

00,02 LIBRARY
      FREEBLO.CKS  <7777>        0040*            32
      SWAPTRA.CKS  <7777>        0600*           384
      OS8    .OS8  <0052>        1200*           640*
      OS8    .RTS  <1152>        3300            1728
      LOGIN  .SAV  <5152>        0011*             9*
      IMAGE  .RK5  <0077>       14540*          6496*
      IMAGE  .RKA  <0077>        6260*          3248*
      IMAGE  .RKB  <0077>        6260*          3248*

FREE BLOCKS:                     6357            3311

OPTION?:
```

In a multiple LISTing of user accounts, they are LISTed in order in which they appear in the MFD.

An acceptable alterative to the previously defined procedures is to input the account information on the same line as the word

LIST. Figure 5-10 contains examples of the multiple command LIST.

Figure 5-10
Multiple Command LIST

```
OPTION?: LIST MFD
OPTION?: LIST [00,01]
```

The above specifications can be used to list the Master file directory. To LIST account number, [00,47], the following alternative is acceptable.

```
OPTION?: LIST 00,47
```

The account number brackets are optional in all input situations.

At installations where security is important, you might wish to inhibit the printing of the account password. Enter NOPASSWORD in response to the "OPTION?" question in order to inhibit the printing of account passwords. Passwords are not printed on any LISTing until you enter PASSWORD in response to the "OPTION?" question. A sample use of these commands is contained in Figure 5-11.

Figure 5-11
Listing Without Passwords

```
OPTION?: NOPASSWORD                          [inhibits printing passwords]
OPTION?: LIST MFD

   00,01                <7777>          0001          1
   00,02                <7777>          0001          1
   00,03                <7777>          0002          2

FREE BLOCKS:                             6357       3311

OPTION?: PASSWORD                            [restores printing passwords]
OPTION?: LIST MFD

   00,01 PASSWORD
   00,01 PASSWORD       <7777>          0001          1
   00,02 LIBRARY        <7777>          0001          1
   00,03 OPERATOR       <7777>          0002          2

FREE BLOCKS:                             6357       3311

OPTION?:
```

At the bottom of each listing, the number of free blocks is printed. This number represents the amount of blocks which are not assigned to ETOS files. Once an ETOS file is created, the number of blocks in the file are subtracted from the amount of free blocks.

If you wish to obtain only the number of free blocks left in the ETOS area, enter EMPTY or MTY in response to the "ACCOUNT?:" question. A list of free blocks is contained in Figure 5-12.

Figure 5-12
List of Free Blocks

```
OPTION?: LIST
ACCOUNT?: EMPTY

FREE BLOCKS:    6357    3311
OPTION?:
```

Each user's number of free blocks after a SYSGEN depends on the

number of swap tracks that is specified. The calculation of the size of the swap tracks file ([Ø,2] SWAPTRA.CKS) is presented in 4.2. On a newly SYSGENed RKØ5 pack, the number of free blocks is 3695 (decimal) minus the size of the swap tracks file. On a newly Sysgened System Industries pack, the number of free blocks is 1Ø255 (decimal) minus the size of the swap tracks file. The amount of initial free blocks can be increased by DELETing scratch blocks (see 5.6) and REDUCing virtual SYS (see 5.8).

5.5 CREATING ACCOUNTS AND ETOS FILES

The "CREATE" option of ACCNT can be used to create accounts or ETOS files under accounts. Section 5.2 explains account attributes.

```
OPTION?: CREATE
ACCOUNT NUMBER?:
```

After you enter CREATE, ACCNT asks for an "ACCOUNT NUMBER?:". If you enter [Ø,1] or MFD, ACCNT knows that you wish to create an account. The system responds with

```
[PROJECT, PROGRAMMER] NUMBER?:
```

Enter the account number of the account you wish to create. The account number is made up of a project number and a programmer number. Each number must be an octal number between ØØ and 77. Make sure that the account number you specify does not already exist in the master file directory. After you have entered the account number, the system responds with

```
NEW PASSWORD IS?:
```

Specify the password of the account you are creating. The password, which will be required at LOGIN time, consists of one

to eight alphanumeric characters. If you do not want the password to echo as you type it in, press the ESCape key or ALT MODE key as the first character of the input line. After you have entered the password, the system responds with

```
PRIVILEGE?:
```

Enter the protection code word of the account you are creating. The protection code controls whether or not the account is privilegable and the amount of memory which is automatically assigned at LOGIN time. See 5.1 for the layout of this word. There are two special words which may be entered in response to this question. If you enter YES, the system assigns the account a protection code of 4Ø77. 4Ø77 is used for a privileged account, which is assigned 8K words at LOGIN time. If you enter NO, the system assigns the account a privilege word of ØØ77. ØØ77 is used for a non-privileged account, which is assigned 8K words at LOGIN time.

After you have entered the privilege word, the system responds with

```
OPTION?:
```

At this point, you may specify another ACCNT option.

A sample creation of an account is contained in Figure 5-13.

Figure 5-13
Sample Account Creation

```
OPTION?: CREATE

ACCOUNT NUMBER?: MFD
[PROJECT, PROGRAMMER] NUMBER?: 10,05
NEW PASSWORD IS ?: TEST
PRIVILEGE?: 3077

OPTION?:
```

All accounts must be created in this manner. CREATE also will accept condensed input. A sample creation of an account with account number [00,47] with a password of SMITH and a 4077 protection code is contained in Figure 5-14.

Figure 5-14
Multiple Command CREATE

```
OPTION?: CREATE
ACCOUNT NUMBER?: MFD [00,47] SMITH 4077
OPTION?:
```

You may or may not desire to have different persons using the same account number. However, you cannot have two users creating OS/8 storage files or two users running COS in the same account. Therefore, you should create accounts so that you will not have two users of OS/8 storage files and/or COS LOGged INto the same account at the same time.

If you wish to create OS/8 or COS storage areas on a pack other than DK0 (see 5.11), you must create the account number also on DK0. The reason for this duplication is that the LOGIN program checks the MFD on DK0 to see if the account exists. If the account existed only DK1, DK2 or DK3, you would not be able to LOGIN to the account in order to access the storage area under the account.

After you have created accounts in the Master file directory, you

can create ETOS files under the account. ETOS files can only be created for accounts that already exist on the ETOS disk(s). Section 5.2 explains ETOS files and attributes.

```
OPTION?: CREATE
ACCOUNT NUMBER?:
```

After you enter CREATE, ACCNT asks for an "ACCOUNT NUMBER?:". If you enter an account number other than [0,1], which has already been created, ACCNT knows that you wish to create an ETOS file in that account. After you have entered the account number, the system responds with

```
FILENAME.EXT?:
```

Enter the name of the ETOS file which you wish to create. For example, if the file will be used as an OS/8 storage area, enter OS8DISK.DSK. If the file will be used as COS program and file storage area, enter COSDISK.RTS. If file will be used strictly as a COS file storage area, enter DATAFI.LES.

If the ETOS file will be used for a different purpose than as an OS/8 or COS storage area, enter the desired name. The filename can be from one to seven alphanumeric characters. The file extension can be from one to three alphanumeric characters. After you have entered the ETOS filename and extension, the system responds with

```
<PROTECTION CODE>?:
```

Page 5-8 0050 or 0074

Enter the file protection code of the ETOS file you are creating. This protection code controls read and write access to the ETOS file. The layout of this code is presented in 5.2.2. The protection code may be entered with or without angle brackets. After you have entered the file protection code, the system

responds with

FILE LENGTH, IN BLOCKS?:

Enter the desired length, in blocks, of the ETOS file you are creating. If the length is not preceded by any characters, it is taken as an octal number. The octal length is double precision, so that the length can be from Ø to 37777777. If the length is preceded by a "D" or a "#", the length is taken as a decimal number. The decimal length can be from Ø to 8,888,608. The specified length should be less than the number of free blocks left on the disk. If the specified length is greater than the number of free blocks and the file is variable length, an error is generated but all available space is assigned to the file. If the specified length is greater than the number of free blocks and the file is fixed length, an error is generated and the file is not created.

After you have specified the file length, the system responds with

FIXED LENGTH, OR VARIABLE?:

Specify either FIXED or VARIABLE. A fixed length file will not be able to be EXTENDed or REDUCEd. If you want to change the size of a FIXED length file, you must DELETE the file and reCREATE it. OS/8 and COS storage allocations are normally CREATEd as VARIABLE length files. This allows you to modify the size of your storage areas according to your changing needs. The only reason for creating a storage area as a FIXED length file is to insure that a hostile user will not be able to EXTEND his area and fill up the disk.

After you enter FIXED or VARIABLE, the system responds with

* LARGE(?) FILES MUST BE VARIABLE (eg D3ØØØØ). Attempting to create fixed length file gives '7=ENTER ERROR' (no room).

```
OPTION?:
```

You can specify another ACCNT option at this time.

A sample creation of an ETOS file is contained in Figure 5-15. Do not follow this example unless you wish to CREATE an ETOS file for some special purpose software.

Figure 5-15
Sample ETOS File Creation

```
OPTION?: CREATE

ACCOUNT?: [01,50]                    (account previously created)
FILENAME.EXT?: TEST.FIL
<PROTECTION CODE>?: <0050>
NEW FILE LENGTH, IN BLOCKS?: D107    (107 decimal blocks)
FIXED LENGTH, OR VARIABLE?: FIXED

OPTION?:
```

If the ETOS file is an OS/8 or COS storage area, there may be unused blocks in the area printed out by the OS/8 "DIR" or COS "DI" command. However, these unused blocks are not part of the list of ETOS free blocks since they cannot be assigned to ETOS files. The number of free blocks listed in an OS/8 or COS directory is only available to the account user; it is not available to the ETOS system and does not appear in the ETOS free block listing.

You should not assign all free blocks to ETOS files. You should leave a minimum of several hundred decimal blocks which can be used for EDUCOMP BASIC temporary files, cumulative statistics files, spooler temporary files, spooler queues, the system status program, the on-line backup program, the installation file and LOGIN messages. You may also write software which utilizes this free space.

see P. 5-18

5.5.1 Creating an OS/8 Storage Area

Creating an OS/8 storage area follows the dialogue for creating an ETOS file.

The ETOS filename must be specified on OS8DISK.DSK (.DSK or .OS8 can be used as abbreviations). The reason this name must be used is that LOGIN looks for a file of this name, which is automatically associated with channel 2.

If you are creating an OS/8 storage area (OS8DISK.DSK), seven blocks will be allocated to the OS/8 directory. You should assign a length of seven blocks more than the storage area you require.

There is one additional step which must be followed when creating an OS/8 storage area. The directory must be initialized. You cannot create any OS/8 files inside the area unless this initialization is performed. First, establish the OS8DISK.DSK following the directions on creating an ETOS file. When ACCOUNT prints the "OPTION?:" question after the file creation, type "ZERO". For example OPTION?:

```
OPTION?: ZERO
ACCOUNT?:
```

Enter the account number of the account which contains the OS8DISK.DSK file. The system responds with

```
FILENAM.EXT?:
```

Enter OS8DISK.DSK to inform ACCNT that you are ZEROing an OS/8 storage area. The area which you are ZEROing must have a file length greater than 7 blocks and less than 4095 blocks. After the storage area is ZEROed, the system responds with

```
OPTION?:
```

You can now specify another ACCNT option.

The operator's account [0,3] should have a small OS/8 storage area, which can be used for creating an installation file or changing the LOGIN message. The dialogue contained in Figure 5-16 creates a 30 (decimal) block OS/8 storage area for account [0,3].

Figure 5-16
Sample Creation of OS/8 Storage Area

```
OPTION?: CREATE

ACCOUNT?: [0,3]
FILENAM.EXT?: OS8DISK.DSK
<PROTECTION CODE> ?: <0074>
NEW FILE LENGTH, IN BLOCKS?: D30
FIXED LENGTH, OR VARIABLE?: VARIABLE

OPTION?: ZERO

ACCOUNT?: [0,3]
FILENAM.EXT?: OS8DISK.DSK

OPTION?:
```

If you are CREATing an OS/8 storage area for an account other than [0,3], you must have first CREATEd the account using the procedures of 5.5. A sample creation of an account and OS/8 file storage, using the multiple command syntax, is contained in Figure 5-17.

Figure 5-17
Multiple Command CREATE

```
OPTION?: CREATE MFD [7,50] OS8ACCT 3077
OPTION?: CREATE [7,50] .OS8 <0050> D200 VARIABLE
OPTION?: ZERO [7,50] .OS8
OPTION?:
```

If you followed the two examples in this section which CREATEd an account [7,50] in the master file directory and an OS/8 storage area in accounts [0,3] and [7,50], entering LIST ALL in response to the "OPTION?:" question produces the list contained in Figure 5-18.

Figure 5-18
LIST After Sample CREATEs

```
00,01 PASSWORD
   00,01      PASSWORD        <7777>      00001        1
   00,02      LIBRARY         <7777>      00001        1
   00,03      OPERATOR        <7777>      00002        2
   07,50      OS8ACCT         <3077>      00001        1  APR. 1 '79

00,02 LIBRARY
      FREEBLO.CKS             <7777>      0040*       32*
      SWAPTRA.CKS             <7777>      0600*      384*
      OS8     .OS8            <0052>      1200*      640*
      OS8     .RTS            <1152>      3300      1728
      LOGIN   .SAV            <5152>      0011*        9*
      IMAGE   .RK5            <0077>     14540*     6496*
      IMAGE   .RKA            <0077>      6260*     3248*
      IMAGE   .RKB            <0077>      6260*     3248*

00,03 OPERATOR
      JOB03   .SBK            <0025>      0030*       24*
      JOB04   .SBK            <0025>      0030*       24*
      JOB05   .SBK            <0025>      0030*       24*
      JOB06   .SBK            <0025>      0030*       24*
      JOB07   .SBK            <0025>      0030*       24*
      JOB10   .SBK            <0025>      0030*       24*
      JOB11   .SBK            <0025>      0030*       24*
      JOB12   .SBK            <0025>      0030*       24*
      JOB13   .SBK            <0025>      0030*       24*
      JOB14   .SBK            <0025>      0030*       24*
      JOB15   .SBK            <0025>      0030*       24*
      JOB16   .SBK            <0025>      0030*       24*
      JOB17   .SBK            <0025>      0030*       24*
      JOB20   .SBK            <0025>      0030*       24*
      JOB21   .SBK            <0025>      0030*       24*
      JOB22   .SBK            <0025>      0030*       24*
      OS8DISK.DSK             <0074>      0036*       30*  APR. 1 '79

07,50 OS8ACCT
      OS8DISK.DSK             <0050>      0310        200  APR. 1 '79

FREE BLOCKS:                              6044       3108
OPTION?:
```

5.5.2 Creating a COS Program Storage Area (omit)

Creating a COS storage area follows the dialogue creating an ETOS file. The ETOS filename must be specified as COSDISK.RTS (.RTS and .COS can be used as abbreviations). The reason this name must be used is that the COSBO program looks for a file of this name, which is automatically associated with channel Ø.

If you are creating a COS storage area (COSDISK.RTS), ninety-six blocks is allocated to the COS monitor and directory. You should assign a length of ninety-six blocks more than the storage area you require.

There are two additional steps which must be followed when creating a COS storage area. After the area is created, you must LOGIN to the account containing the storage area and transfer COS to the area with the dialogue contained in Figure 5-19.

Figure 5-19
Initializing a COS Storage Area

```
.^VS                          (exit OS/8 to SCALE)
!CLOSE 3
!LOOKUP 3=COSDISK.RTS         (associate COS area with Channel 3)
!BOOT                         (return to OS/8)
.R COSBLD
'COSBLD' V1.Ø12

CHANNEL NUMBER? 3
ZERO OUTPUT DIRECTORY? YES
COS SYSTEM BUILT
.
```

Refer to Chapter 7 of the User's Guide for information on channels. The above procedure associates the COS storage area with channel 3 and transfers COS to it. If you are using channel 3 for another purpose, use another channel in the dialogue for the CLOSE, LOOKUP and CHANNEL NUMBER. You must enter YES in response to the "ZERO OUTPUT DIRECTORY?" question. This response

causes the program to initialize the COS directory.

If you had corrupted your copy of COS but not the COS programs, you can run COSBLD and enter NO in response to the "ZERO" question. This entry allows you to restore COS without initializing the directory.

After you have transferred COS into the COS storage area, the last step involves transferring COS system programs to the area. With the exception of BREAK, OS8BO, SYSGET and UNBRK, these programs are unmodified COS300 programs. If you overlay them with copies from your stand-alone COS system, the programs will still operate properly. The dialogue contained in Figure 5-20 copies the COS system programs from virtual OS/8 SYS to a COS system area. This dialogue assumes that you have just followed the dialogue which transfers COS.

Figure 5-20
Copying System Programs into a COS Storage Area

```
.COPY CHN3:*.SV<SYS:*.CS/Q

FILES COPIED:

BREAK.CS ? Y
BUILD.CS ? Y
COMP.CS  ? Y
CONVEX.CS? Y
CREF.CS  ? Y
DAFT.CS  ? Y
DFDIR.CS ? Y
LINCHG.CS? Y
OS8BO.CS ? Y
PATCH.CS ? Y
PIP.CS   ? Y
PRINT.CS ? Y
SORT.CS  ? Y
SYSGET.CS? Y
UNBRK.CS ? Y
UPDATE.CS? Y
.
```

The COS programs are stored on virtual OS/8 SYS with the

extension ".CS". Most of the programs are unmodifed COS CUSPS. The use of all of these programs is documented in 1.7. After you enter the COPY command in the previous dialogue, the system prints each program name with a question mark. If you wish to transfer the program to your COS storage area, enter Y. If you do not wish to transfer the program, enter N. This procedure allows you to copy only the programs which you need. You should always copy PIP to the storage areas. PIP is needed to transfer your DIBOL programs from your stand-alone COS disks to your ETOS COS storage area. This procedure is documented in section 5.10.2.

There are two methods you can follow for initializing COS accounts. You may create all of your COS storage areas with ACCNT and then initialize all areas by LOGging IN, performing the initializing procedures and LOGging OUT. The other method is to create a COS storage area with ACCNT, exit ACCNT, initialize the area by LOGging IN, performing the initializing procedures and LOGging OUT. This procedure would be followed for each COS storage area. The first method is the most efficient of the two methods, but the second method insures that you initialize each account. You may use either method.

After you have initialized a COS storage area by transferring COS and the COS programs to it, you can access the area by LOGging into the account containing the storage area and entering R COSBO. The COSBO program closes channel 0 and associates channel 0 with the file COSDISK.RTS. After the program is run, you are running ETOS COS. All commands operate basically as they do under stand-alone COS. See Chapter 5 in the System User's Guide for a list of differences between stand-alone COS and ETOS COS.

The sample creation of a 200 (decimal) COS storage area for account [10,21] is contained in Figure 5-21. This dialogue assumes that account [10,21] has already been created with a

password of COSACCT.

Figure 5-21
Sample Creation of COS Program Storage Area

```
OPTION?: CREATE

ACCOUNT?: [10,21]
FILENAM.EXT?: COSDISK.RTS
<PROTECTION CODE>?: <0050>
FILE LENGTH, IN BLOCKS?: D200
FIXED LENGTH, OR VARIABLE?: VARIABLE

OPTION?: EXIT

CHANNELS 5,6,7 CLOSED
.LOGOUT
JOB 03 LOGGED OUT ON CONSOLE 00
3:52:08 P.M. ON FRI 06-JUL-79
ELAPSED TIME:
58 MINUTE(S) 18.7 SECONDS
GOOD AFTERNOON
!

!LOGIN
ETOS V5B AT 03:52:11 P.M. ON FRI 06-JUL-79
ACCOUNT? 10,21
PASSWORD? COSACCT
JOB 03 LOGGED IN ON CONSOLE 00

.^VS
!LOOKUP 3=COSDISK.RTS
!CONTINUE
R COSBLD
'COSBLD' V1.012

CHANNEL NUMBER? 3
ZERO OUTPUT DIRECTORY? YES
COS SYSTEM BUILT
.COPY CHN3:*.SV<SYS:*.CS/Q

FILES COPIED:
COMP.CS? Y
BREAK.CS? N
BUILD.CS? N
CONVEX.CS? Y
CREF.CS? N
DAFT.CS? Y
DFDIR.CS? Y
```

Figure 5-21 (continued)

```
LINCHG.CS? N
OS8BO.CS? Y
PATCH.CS? N
PIP.CS? Y
PRINT.CS? N
SORT.CS? Y
SYSGET.CS? Y
UNBRK.CS? N
UPDATE.CS? N
.R COSBO
COS 3.Ø7GB
.
```

5.5.3 Creating a COS Data File Storage Area

Creating a COS data file storage area follows the dialogue for creating an ETOS file. The ETOS file name must be specified as DATAFI.LES. The reason this name must be used is that SYSGET (ETOS version of SYSGEN) looks for files of this name for data files. Unlike stand-alone COS, data files should not be stored in the same area as programs. Data files should be stored in the DATAF1.LES areas. The reason the data files should be separate is that COS assigns data files sequentially on disk areas, which are separate from the program area. If you have data files stored on a program area, COS starts the files at a block which is equal to the length of the storage area minus the total length of the files. If you change the size of the storage area at a later date, all of your files must be moved.

The protection code of the data file storage area should be determined by deciding which users will access the area. It is valid to have two users accessing the same file area at the same time. However, since there is no record lockout feature, you must be careful of simultaneous access. If two users update the same record in the same file at the same time, one user's changes will be lost. You must insure that two users don't access the same record by incorporating a record lockout feature in your applications programs or by scheduling your processing so that

more than one user will never be updating the same record.

Note: You can have any number of users printing reports on the same file.

The length of the data file area can be computed by adding up the total number of segments in all of the files you want to transfer into the area. Multiply the total number of segments by sixteen to obtain the number of decimal blocks required for the area. Data file areas should normally be created as variable length ETOS files, in case you wish to change the length of a COS data file or add data files to the storage area.

Creation of COS data file areas are performed via the ACCNT CREATE option. Unlike OS/8 storage areas or COS program storage areas, there are no additional steps which are necessary to create a COS data file storage area. A sample creation of a COS data file area containing 106 segments (1696 decimal blocks) in account [10,21] is listed in Figure 5-22. This example assumes that account [10,21] has been previously created.

Figure 5-22
Sample Creation of COS Data File Storage Area

```
OPTION?: CREATE
ACCOUNT?: [10,21]
FILENAM.EXT?: DATAF1.LES
<PROTECTION CODE>?: <0050>
FILE LENGTH, IN BLOCKS?: D1696
FIXED LENGTH, OR VARIABLE?: VARIABLE
OPTION?:
```

A multiple command syntax which is equivalent to the dialogue of Figure 5-22 is contained in Figure 5-23.

Figure 5-23
Multiple Command CREATE

```
OPTION?: CREATE [10,21] DATAF1.LES <0050> D1696 VARIABLE
OPTION?:
```

5.6 DELETING ACCOUNTS AND ETOS FILES

Deleting ETOS files and accounts follows the same general format as the creation process. There is one major difference. In order to create ETOS file, a user account must be created beforehand. When deleting files and accounts, all the files within the account must be deleted first. When all files within the account are deleted, then the account may be deleted.

```
OPTION?: DELETE
ACCOUNT NUMBER?:
```

After you enter DELETE, ACCNT asks for an "ACCOUNT NUMBER?:". Enter the account number which contains ETOS files, which you wish to delete. The system responds with

```
FILENAM.EXT?:
```

Enter the name of the ETOS file you wish to delete. If the file is an OS/8 storage area, enter OS8DISK.DSK. If the file is a COS program storage area, enter COSDISK.RTS. If the file is a COS data file storage area, enter DATAF1.LES. If you wished to delete scratch blocks (see 5.15.3), enter JOBXX.SBK, where XX represents the job number.

After you have entered the ETOS file name, the system responds with

```
OPTION?:
```

At this point, you may specify another ACCNT option.

An example of an ETOS file deletion of the OS/8 storage area for account [00,27] is contained in Figure 5-24.

Figure 5-24
Sample ETOS File Deletion

```
OPTION?: DELETE
ACCOUNT NUMBER?: [00,27]
FILENAME.EXT?: OS8DISK.DSK
OPTION?:
```

After you have deleted all files under an account you can delete the account from the master file directory.

```
OPTION?: DELETE
ACCOUNT?: MFD
```

If you enter [0,1] or MFD, ACCNT knows that you wish to delete an account. The system responds with

```
[PROJECT,PROGRAMMER] NUMBER?:
```

Enter the account number of the account which you wish to delete. After you have entered the account number, the system responds with

```
OPTION?:
```

At this point, you may specify another ACCNT option.

A sample deletion of ETOS account number [0,27] is contained in Figure 5-25.

Figure 5-25
Sample Account Deletion

```
OPTION?: DELETE
ACCOUNT NUMBER?: [00,01] or ACCOUNT or MFD
[PROJECT, PROGRAMMER]NUMBER?: [00,27]
```

An example of a multiple command syntax is contained in Figure 5-26. This example performs the same operation as the dialogue

contained in Figure 5-24 and Figure 5-25.

Figure 5-26
Multiple Command DELETE

```
OPTION?: DELETE [00,27]OS8DISK.DSK
OPTION?: DELETE MFD [00,27]
OPTION?:
```

Normally only alphanumeric characters are accepted in file names. However file names sometimes become corrupted and do contain non-alphanumeric characters. The SPECIAL mode of operation will permit you to enter these non-alphanumeric characters and have them accepted. For example,

```
OPTION?: SPECIAL
OPTION?:
```

You may proceed to delete the ETOS file with non-alphanumeric characters.

After doing this, you should return the ACCOUNT program to normal operation by typing NORMAL at the next option call. For example,

```
OPTION?: NORMAL
```

5.7 CHANGING ACCOUNT AND ETOS FILE ATTRIBUTES

The RENAME option of the ACCOUNT program allows the changing of the account password and protection code.

```
OPTION?: RENAME
ACCOUNT NUMBER?:
```

Enter MFD or [00,01] to indicate that you wish to change an account password or protection code. The system responds with

[PROJECT, PROGRAMMER] NUMBER?:

Enter the account which you wish to modify. The system responds with

NEW PASSWORD IS?:

Enter the new password for the specified account. If you do not wish to change the password, you must enter the old password. The system responds with

PRIVILEGE?:

Enter the new protection code for the specified account. If you do not wish to change the protection code, you must enter the old protection code. The system responds with

OPTION?:

At this point, you may specify another ACCNT option.

A sample use of the RENAME option to change the password of account [00,27] to FUN and the protection code to 3077 is contained in Figure 5-27.

Figure 5-27
Renaming Account Attributes

```
OPTION?: RENAME
ACCOUNT NUMBER?: MFD  (or [00,01])
[PROJECT,PROGRAMMER]NUMBER?: [00,27]
NEW PASSWORD IS?: FUN
PRIVILEGE?: 3077
OPTION?:
```

The RENAME option can also be used to change ETOS file names and file protection codes.

```
OPTION?: RENAME
ACCOUNT NUMBER?:
```

Enter the account number containing the ETOS file which you wish to modify. The system responds with

```
OLD FILENAM.EXT?:
```

Enter the name of the ETOS file which you wish to modify. The system responds with

```
NEW FILENAM.EXT?:
```

Enter the new name of the ETOS file. If you do not wish to change the file name, enter the old file name. The system responds with

```
<PROTECTION CODE>?:
```

Enter the new file protection code for the specified ETOS file. If you do not wish to change the protection code, enter the old protection code. The system responds with

```
OPTION?:
```

At this point, you may specify another ACCNT option.

A sample use of the RENAME option to change the ETOS file OS8DI.DSK in account [00,27] to the name OS8DISK.DSK and a protection code of <0050> is contained in Figure 5-28.

Figure 5-28
Renaming ETOS File Attributes

```
OPTION?: RENAME
ACCOUNT NUMBER?: [00,27]
OLD FILENAM.EXT?: OS8DI.DSK
NEW FILENAME.EXT?: OS8DISK.DSK
<PROTECTION CODE>?: 0050
OPTION?:
```

An example of a multiple command RENAME is presented in Figure 5-29. The effect of this dialogue is identical to the effect of the dialogue presented in Figure 5-27 and Figure 5-28.

Figure 5-29
Multiple Command RENAME

```
OPTION?: RENAME MFD [00,27] FUN 3077
OPTION?: RENAME [00,27] OS8DISK.DSK 0050
```

5.8 CHANGING ETOS FILE LENGTHS

Sometimes it may be necessary to lengthen or shorten the size of an ETOS file. The EXTEND command of the ACCOUNT program will do this. For example, the OS/8 storage area for account [00,27] may be too small to run the necessary program and need to be extended. Contrawise, the OS8 storage area may not be fully used and there is demand from other users for storage, or for free blocks within the system. The storage area may be reduced. The same EXTEND command will do this operation. The REDUCE option is identical to the EXTEND option. Either option may be used to lengthen or shorten the size of an ETOS file.

```
OPTION?: EXTEND
ACCOUNT NUMBER?: [00,27]
```

Enter the account number which contains the storage area you wish to extend or reduce. The system responds with

FILENAM.EXT?:

Enter the name of the storage area you wish to extend or reduce. Enter OS8DISK.DSK for an OS/8 storage area. Enter COSDISK.RTS for a COS program storage area. Enter DATAFI.LES for a COS data file storage area. Enter [0,2] for the account number and OS8.RTS for the file name if you wish to change the size of the virtual OS/8 system device. The file which you specify must be a variable length file. If you wish to change the length of a fixed length file, you must delete the file and recreate it. If you do not wish to lose the contents of the fixed length file, create an ETOS file of the desired length. Transfer the contents of the old file to the new file and delete the old file. After you have specified the ETOS file name, the system responds with

FINAL FILE LENGTH, IN BLOCKS?:

Enter the desired file length of the specified ETOS file. If the length is not preceded by any characters, it is taken as an octal number. If the length is preceded by a "D" or a "#", the length is taken as a decimal number. If you reduce the length of the file, insure that you do not reduce valid OS/8 or COS files. It is advisable to run PIP option S on OS/8 storage areas or PIP option E on COS program storage areas before reducing the size of these areas. After you have entered the file length, the system responds with

OPTION?:

A sample of changing an OS/8 storage area in account [00,27] to a length of 107 (base 10) blocks is contained in Figure 5-27.

Figure 5-30
Sample Use of ACCNT EXTEND Option

```
OPTION?: EXTEND
ACCOUNT NUMBER?: [00,27]
FILE NAME.EXT?: OS8DISK.DSK
FINAL FILE LENGTH, IN BLOCKS?: D107
OPTION?:
```

A sample use of the multiple command EXTEND is presented in Figure 5-31. This command has the same affect as the dialogue contained in Figure 5-30.

Figure 5-31
Multiple Command EXTEND

```
OPTION?: EXTEND [00,27] OS8DISK.DSK D107
OPTION?:
```

If the ETOS file you have modified is an OS/8 storage area (OS8DISK.DSK) or a COS storage area (COSDISK.RTS), there is one additional step which must be followed. After you have performed the EXTEND, ETOS is aware of the change in the size of the storage area. However, OS/8 or COS is not aware of the change in storage. The OS/8 or COS directory must be updated to reflect the current size. To update a directory, you must exit the ACCNT program, LOGOUT of the current account and LOGIN to the account which contains the storage area. If you are updating an OS/8 directory, follow the dialogue presented in Figure 5-32.

Figure 5-32
Updating an OS/8 Directory

```
.SQU DSK:/O or SQU SYS:/O      (OS/8 storage area or OS/8
                                system device)
.                              (file size now changed)
```

DO NOT SQUISH THE DIRECTORY OF A STAND/ALONE PACK. USE SQUASH.

If you are updating a COS directory, follow the dialogue presented in Figure 5-33.

Figure 5-33
Updating a COS Directory

```
.R COSBO                       (enter COS)

COS MONITOR 3.07GB

.ER
.R SYSGET

DONE
.                              (file size now changed)
```

5.9 ADDITIONAL ACCNT INFORMATION

Sections 5.3 through 5.8 contain all the information necessary to run the ACCNT program with standard options. Sections 5.9 contains extended options which allow more efficient use of the program. This section also contains an explanation of restrictions and possible errors and how to eliminate some of these restrictions. Some of the information in this section requires sophisticated knowledge about the system. If you do not understand any of the information presented in the section, you may simply skip that particular discussion.

5.9.1 Input Conventions

During input of information pertaining to an option, there are several conventions which should be remembered.

1) All lower-case characters are converted automatically to upper case.

2) All non-alphanumeric characters terminate an input specification.

3) Entering CTRL/O, while the system is listing, suppresses output until input is required.

4) Entering CTRL/P returns you to the OPTION? question. If you enter CTRL/P before you read the initial OPTION? question, the "ETOS DEVICE NAME?: question is repeated.

5) Entering CTRL/S stops output until a CTRL/Q is entered from the keyboard.

6) Entering RUBOUT deletes the last input character on the current line. Entering CTRL/U deletes the entire input line.

7) Entering the ESCape or ALT MODE keys disables echoing of input until another ESCape key or ALT MODE key is input or a carriage return is input. This option allows you to enter sensitive data, such as an account password without allowing anyone else to read this data.

8) All options may be reduced to their first four letters.

9) Some options require several user input specifications. For example, creation of an account requires input of an account number, a password and an account protection code. The system prompts you for each specification. After you input the desired value for the item, press the carriage return key on the terminal. Using this process, entering three specifications requires three lines of input. As an alternative to this mode of input, you can enter multiple specifications on one line, separated by a space. A carriage return is used to terminate the input. Any input which is required is requested by the system. An example of equivalent specifications, using different numbers of input lines is contained in Figure 5-34.

Figure 5-34
Equivalent ACCNT Specifications

```
OPTION?: CREATE
ACCOUNT?: 1
[PROJECT, PROGRAMMER] NUMBER?: 17,20
PASSWORD?: DEMO
PROTECTION?: 5077
OPTION?

OPTION?: CREATE 1 17,20
PASSWORD?: DEMO 5077
OPTION?:

OPTION?: CREATE 1 1720 DEMO 5077
```

Extra input may be specified at the end of a command. This input is treated as a comment. This commenting is expecially valuable when commands come from an OS/8 file. A sample use of a comment is contained in Figure 5-35.

Figure 5-35
Sample Use of Comment

```
OPTION?:    CREATE 1 1720 DEMO 5077 BECAUSE I NEED THIS
```

10) Account numbers may be input with or without commas or square brackets. The account number specifications contained in Figure 5-36 are equivalent.

Figure 5-36
Equivalent Account Number Specifications

```
1
0,1
00,01
[0,1]
```

11) All OS/8 storage allocations should be specified as OS8DISK.DSK. LOGIN looks for a file with this name. Acceptable abbreviations for "OS8DISK.DSK" are ".OS8" and ".DSK".

12) All COS storage allocations should be specified as COSDISK.RTS. COSBO looks for a file with this name. Acceptable abbreviations for "COSDISK.RTS" are ".COS" and ".RTS".

13) Account [0,1] is the master file directory. The use of this account is explained in 5.2.1. Acceptable designations for this account in addition to "[0,1]" are

"MFD" and "ACCOUNT".

14) ETOS filenames must consist entirely of alphanumeric characters. The ACCNT program normally rejects filenames with non-alphanumeric characters. However, if a file was created with invalid characters by a system or program malfunction, you might want to RENAME or DELETE the file. If you enter SPECIAL in response to the "OPTION?:" question, the ACCNT program displays or accepts all ETOS filenames as specified. The file OS8.RTS is displayed as "OS8@@@@RTS", since "@" represents a null. All input is taken at face value. Multiple commands on one line is not allowed. Abbreviations such as ".OS" or "MFD" are not allowed. Input is taken from the keyboard, even if an input file is specified. The reason for this input convention is that "SPECIAL" is only used after some malfunction. Malfunctions should be fixed from the keyboard only. To re-enable the usual mode of input and display, enter NORMAL in response to the "OPTION?" question.

15) Any abbreviations (e.g., "ACCOUNT", ".OS8") may be specified by their first two letters.

5.9.2 Restrictions

The ACCNT program protects you from performing operations which would corrupt the structure of your ETOS disk. A list of the operations, which are prohibited, follows.

1) EXTENDing or REDUCing a fixed length file or an account entry in the master file directory.

2) DELETing, RENAMing, EXTENDing, REDUCing, or ZEROing the ETOS files [Ø,2]IMAGE.RK5, [Ø,2]IMAGE.RKA, [Ø,2]IMAGE.RKB, [Ø,2]IMAGE.RKC, [Ø,2]IMAGE.RKD, AND [Ø,2]FREEBLO.CKS. In addition to these files, you should not delete [Ø,2]SWAPTRA.CKS, [Ø,2]OS8.OS8 and [Ø,3]JOBØ3.SBK if the ETOS device name you specified is DKØ.

3) ZEROing an ETOS file less than 8 or greater than 4095 blocks long or ZEROing an account entry in the Master file directory.

4) DELETing an account from the Master file directory which contains ETOS files.

5) Running ACCNT in any account other than [Ø,3] or on any

terminal other than the console terminal. This restriction may be bypassed with the information given in 5.9.4 and 5.9.5.

5.9.3 ACCNT Error Messages

If a specification error occurs after an input line, ACCNT prints the line up to the point that the error occurred and then prints the error message. This tells you what part of the input caused the error message. A sample error is contained in Figure 5-37.

Figure 5-37
Sample ACCNT Error

```
OPTION: LOST
LO
**ILLEGAL OPTION WORD
OPTION?
```

ACCNT returns to the question and asks it again. If an OS/8 input file is specified, the run is aborted and ACCNT returns to the OS/8 monitor. If multiple commands are specified, all valid commands before the error are accepted and the question corresponding to the item in error is printed. You can finish the specification or enter CTRL/P to return to the "OPTION?:" question. It is safer to enter CTRL/P because if an error occurred, other information may be in error also.

All errors output by the ACCNT program are listed below.

```
ACCNT.SV NOT FOUND
```

The ACCNT program was run by a non-privileged user or from an invalid console or account. The only acceptable console on the distribution system is console ØØ. The only acceptable account on the distribution system is account [Ø,3].

```
** ACCOUNT NOT FOUND
```

You are attempting to create or access an ETOS file in an account which does not exist. Every ETOS file which is created must have had an account created for it in the master file directory.

** ALREADY EXISTS

An account number has been specified as a new entry to the Master file directory, which already exists in the MFD.

** BAD INPUT! RE-ENTER

The form of the input is incorrect. Check the documentation for the command for the proper form.

** BAD PASSWORD OR FILENAME! RE-ENTER

An account password or ETOS filename was specified with non-alphanumeric characters.

** DANGEROUS OR IMPOSSIBLE COMMAND!**

A command was entered with invalid arguments. Section 5.3.2 contains an explanation of all valid commands.

X = DELETE ERROR

A DELETE error occurred while attempting to DELETE an account number from the Master file directory or an ETOS file from an account. "X" corresponds to the errors in the DELETE CHANIO instruction (see 7.4.9, System User's Guide).

DEVICE FULL

The output file specified in the initial ACCNT question has filled the allotted space on the output device.

X = DIRECTORY WRITE ERROR

A write error occurred while attempting to zero an OS/8 storage area (OS8DISK.DSK). "X" corresponds to the errors in the WRITEW CHANIO instruction (see 7.4.3, System User's Guide).

X = 'ENTER' ERROR

An ENTER error occurred while attempting to create an account in the Master File Directory or an ETOS file in an account. "X" corresponds to the errors in the ENTER CHANIO instruction (see 7.4.5, System User's Guide).

X = EXTEND/REDUCE ERROR

An EXTEND error or a REDUCE error has occurred when attempting to change the length of an ETOS file. "X" corresponds to the errors in the EXTEND CHANIO instruction (see 7.4.7, System User's Guide) or the REDUCE CHANIO instructions (see 7.4.8, System User's Guide). You can determine which error section to check by whether you are attempting to extend or reduce the size of the ETOS file.

** FIXED LENGTH!

An attempt was made to EXTEND or REDUCE a fixed length ETOS file. A fixed length file may only be CREATED or DELETED.

X = GFILE 'READ' ERROR

An error occurred while READing the user file directory for the account in which the specified file resides. "X" corresponds to the errors in the READW CHANIO instruction (see 7.4.1, System User's Guide).

** ILLEGAL 'OPTION' WORD

The option specified is not a valid ACCNT option. See 5.3.2 for a list of options.

X = INQUIRE ERROR

An INQUIRE error occurred while LISTing out the ETOS file structure. An INQUIRE is performed to obtain the ETOS file lengths and dates. "X" corresponds to the errors in the INQUIRE CHANIO instruction (see 7.4.14) System User's Guide).

X = LOOKUP ERROR

ACCNT could not LOOKUP the account or ETOS file specified. The most common cause of this error is the specification of an account or file which does not exist. "X" corresponds to the errors in the LOOKUP CHANIO instruction (see 7.4.6, System User's Guide).

** NO -- DELETE FILES FIRST

An attempt was made to delete an account which contained one or more ETOS files. You must delete all ETOS files under the account before deleting the account.

X = READ ERROR

A READ error occurred while reading the specified ETOS file. "X" corresponds to the errors in the READW CHANIO command (see 7.4.1, System User's Guide).

X = RENAME ERROR

A RENAME error occurred while renaming an account password or

privilege word or while renaming an ETOS file. "X" corresponds to the errors in the RENAME CHANIO command (see 7.4.10, System User's Guide).

(TOO LARGE FOR OS-8 USE. CAUTION...)

The ETOS file being created is larger than 4095 decimal blocks. This size is invalid if the file is an OS/8 storage area (OS8DISK.DSK) or a COS program storage area (COSDISK.RTS). A COS file area might want to use more than 4095 blocks. This message is strictly a warning message. It is not a fatal error.

U.S.R. FAILURE # X

ACCNT cannot find the OS/8 input file or close the OS/8 output file specified in the inital question. "X" corresponds to the U.S.R. argument number. See OS/8 Software Support Manual for further details.

WRITE ERROR

ACCNT cannot write to the specified OS/8 output device. Make sure that the device is write enabled.

5.9.4 Enabling Other Consoles to Run ACCNT

ACCNT keeps a table of four consoles which are allowed to run the program. If an attempt is made to run ACCNT on a console, which is not in the table, ACCNT prints the message "ACCNT.SV NOT FOUND". This message is intentionally misleading for security purposes. A console number of 7777 is used in positions which are unused. The current contents of the table are 0000, 7777, 7777, and 7777. The distribution system allows access of ACCNT only from console 00.

The table of allowable consoles is stored at locations 13403 - 13406 in the ACCNT program. The OS/8 program FUTIL is used to modify the four table entries as illustrated in Figure 5-38.

Figure 5-38
Enabling Other Consoles to Run ACCNT

```
.^VPRIV 4040
^VWE 0
R FUTIL
FILE ACCNT
ACCNT.SV  ssss-eeee 0020 (0016) b.111  29-MAY-79
SET MODE SAVE
0.13403/0000        WWWW<LF>       (Allowable Console #1)
0000.13404\7777     XXXX<LF>       (Allowable Console #2)
0000.13405\7777     YYYY<LF>       (Allowable Console #3)
0000.13406\7777     ZZZZ<RET>      (Allowable Console #4)
WRITE
EXIT
.^VWL0
^VPRIV 40
```

For example, to enable consoles 00,05 and 10, use the dialogue contained in Figure 5-39.

Figure 5-39
Sample Enabling of Consoles

```
.^VPRIV 4040
^VWE 0
R FUTIL
FILE ACCNT
ACCNT.SV  ssss-eeee 0020 (0016) b.111  29-MAY-79
SET MODE SAVE
13403/ 0000  0000<LF>
0000.13404\ 7777 0005<LF>
0000.13405\ 7777 0010<LF>
0000.13406\ 7777 7777<RET>
WRITE
EXIT
.
```

If you have someone other than the system manager start ETOS in the morning, it might be desirable to disable console 0 from being able to run ACCNT.

5.9.5 Enabling Other Accounts to Run ACCNT

ACCNT keeps a table of eight account numbers which you can LOGIN to in order to run the ACCNT program. If an attempt is made to run ACCNT in an account which is not in the table, ACCNT prints the message "ACCNT.SV NOT FOUND". This message is intentionally misleading for security purposes. An account number of 0000 is used in positions which are unused. The current contents of the table are 0003, 0000, 0000, 0000, 0000, 0000, 0000 and 0000. The distribution system allows access of ACCNT only from account [0,3]. Note that account numbers in the table do not contain commas or brackets.

The table of allowable accounts is stored at locations 13410-13417 in the ACCNT program. The OS/8 program FUTIL is used to modify the eight table entries as illustrated in Figure 5-40.

Figure 5-40
Enabling of Other Accounts to Run ACCNT

```
.^VPRIV 4040
^VWE 0

R FUTIL
FILE ACCNT
ACCNT.SV  ssss-eeee  0020 (0016) B.111  29-MAY-79
SET MODE SAVE
13410/0003          aaaa<LF>      (Allowable Account #1)
0000.13411\0000     bbbb<LF>      (Allowable Account #2)
0000.13412\0000     cccc<LF>      (Allowable Account #3)
0000.13413\0000     dddd<LF>      (Allowable Account #4)
0000.13414\0000     eeee<LF>      (Allowable Account #5)
0000.13415\0000     ffff<LF>      (Allowable Account #6)
0000.13416\0000     gggg<LF>      (Allowable Account #7)
0000.13417\0000     hhhh<LF>      (Allowable Account #8)
WRITE
EXIT
.
```

For example, to enable account numbers [0,3], [0,4], [10,5], [77,70] and [5,5], use the dialogue contained in Figure 5-41.

Figure 5-41
Sample Enabling of Accounts

```
.^VPRIV 4040
^VWE 0
R FUTIL
FILE ACCNT
ACCNT.SV  ssss-eeee 0020 (0016) b.111  28-MAR-79
SET MODE SAVE
13410/ 0003  0003<LF>
0000.13411\ 0000  0004<LF>
0000.13412\ 0000  1005<LF>
0000.13413\ 0000  7770<LF>
0000.13414\ 0000  0505<LF>
0000.13415\ 0000  0000<LF>
0000.13416\ 0000  0000<LF>
0000.13417\ 0000  0000<RET>
WRITE
EXIT
.
```

If you have someone other than the system manager start ETOS in the morning, it might be desirable to disable account [0,3] from being able to run ACCNT.

5.9.6 Specifying ACCNT Commands From an OS/8 Input File

The ACCNT program has the facility of accepting commands from an OS/8 ASCII file. The file name is specified when the system prints

```
*OUTPUT<INPUT
*
```

The default extension for the input file is ".AI". This OS/8 file must have been created before the run of the ACCNT program. If you wish to create the file on an ETOS disk, you must have created an OS/8 storage area for the account in which you are running ACCNT. Section 5.5.1 explains the creation of an OS/8 storage area. The format of the input file is exactly the same as if you were specifying the commands from the terminal. When

you run ACCNT and specify the input file, all prompts are printed on the output device as a sign of which commands have been processed. If the output device is the terminal, only the prompts are printed. The actual input is not printed. Enter CTRL/O if you do not wish to see the prompts. If the output device is not the terminal, the prompts and the input from the file are printed. Comments may also be included in the input file. Any input line which begins with a slash (/) is considered a comment and is completely ignored. Also remember that any text after the end of an input specifciation is taken as a comment. If the last line in the file is "EXIT", the program returns to the OS/8 monitor at the termination of the input file. If the last line is not "EXIT", the system RESTARTS the program at the asterisk (*). You may specify an input and output file or enter CTRL/C to exit to the OS/8 monitor or press <RET> to use TTY: input and output.

In Figure 5-42, a sample creation and execution of an input file is contained, which CREATEs an account [01,70] in the Master file directory and an OS/8 storage area for the account and CREATEs an account [02,50] in the Master File Directory and a COS storage area for that account. This example assumes that there is an OS/8 storage area in the account which you are LOGged INto.

Figure 5-42
Sample ACCNT Input File

```
.CREATE ACCTNP.AI
#A
DK0   (System disk)
PASSWORD                    (MFD account password)
CREATE                      (option)
[0,1]                       (create entry in MFD)
[0,70]                      (account number)
OSRUN                       (password)
5077                        (privilege word)
CREATE                      (option)
[00,70]                     (create file in account)
```

Figure 5-42 (continued)

```
OS8DISK.DSK                    (OS/8 storage area)
0050                           (protection code)
1000                           (length = 1000 octal blocks)
VARIABLE                       (variable length file)
CREATE [0,1] [02,50] COSRUN 3077
CREATE [02,50] COSDISK.RTS 0050 1200 VARIABLE
/THE PREVIOUS TWO LINES UTILIZE
/THE MULTIPLE COMMAND SYNTAX (or long line syntax)
EXIT
<CTRL/L>
#E
.^VPRIV 4040
R ACCNT
ACCNT, VERSION 5.53
*OUTPUT<INPUT
*<ACCTNP.AI

ETOS DEVICE NAME?: PASSWORD?:
OPTION?:.[PROJECT,   PROGRAMMER]   NUMBER?:   PASSWORD?:
     PRIVILEGE?:
OPTION?: ACCOUNT?: FILENAM.EXT IS?: < PROTECTION CODE?: NEW
     FILE LENGTH, IN
OPTION?:
OPTION?:
OPTION?:
CHANNELS 5,6,7 CLOSED
CHANNEL 5.6.7 CLOSED
.
```

If you are using an OS/8 input file to specify commands to ACCNT, a RESTART is automatically performed at the end of the input file.

5.9.7 Printing Output to an OS/8 Output File

The ACCNT program has the capability to print all input and output to an OS/8 ASCII file or an output device (e.g., LPT:) the default extension on the output file is ".AO". The input may come from the keyboard or an input file. Each run of the ACCNT program should be saved on paper, so that the output can be used as a reference if a system disk needs recreating. If the output is saved in a file, minor editing can convert the output file into an input file acceptable to the ACCNT program. It is for this reason that no prompts are printed to the output file.

Prompts are always printed on the terminal. Sample specifications in response to the asterisk are contained in Figure 5-43.

Figure 5-43
Sample ACCNT Output Specifications

```
*LPT:< or
*ACCOUT.AO<ACCINP.AI
```

A sample execution of the creation of an input file and an execution of ACCNT with an input and output file is contained in Figure 5-44. This write-up assumes that you have an OS/8 storage area (OS8DISK.DSK) in the account which you are LOGged INto.

Figure 5-44
Sample Use of ACCNT Input and Output Files

```
.CREATE LIST.AI

#A
DK0                              (ETOS system device)
PASSWORD                         (MFD password)
LIST                             (option)
0,5                              (account number)
EXIT                             (exit program)
<CTRL/L>
#E
.^VPRIV 4040
R ACCNT

ACCNT, VERSION 5.53

*OUTPUT<INPUT
*LIST.AO<LIST.AI
ETOS DEVICE NAME?: PASSWORD?:
OPTION?: ACCOUNT?:
OPTION?:
CHANNELS 5,6,7 CLOSED
.
```

If you list the output file (LIST.AO) via the TYPE command, the output contained in Figure 5-45 is printed.

Figure 5-45
Sample ACCNT Output File

```
DKØ
PASSWORD
LIST
Ø,5

ØØ,Ø5 TRS
    ØS8DISK.DSK <ØØ5Ø> Ø6ØØ 384 1Ø MAR. '78

    FREE BLOCKS:        Ø22Ø 144
EXIT
```

If you are inputing commands from the terminal and sending output to a line printer, the line printer output is behind the terminal inut due to buffering. This buffering can be annying if you wish to see the result of a LIST or some other ACCNT option. To force ACCNT to dump the buffer on the output device, enter WRITE in response to the "OPTION?:" question. The program fills the buffer with nulls to force the buffer to be output. These nulls are ignored by the output device.

5.1Ø ACCESSING A PRIVATE PACK

ETOS has the capability to access standard OS/8 or COS format system or data disks. These packs are known as private packs. Since these packs have OS/8 or COS directories, only one user can update the directory of a private pack at a time. If the pack is a COS data pack, any number of users could update information on the disk, because data files are not directory-oriented. In order to be safe, you should not allow access of a COS or OS/8 program pack by more than one user.

Private packs are usually used only to retrieve data from a stand-alone system or to maintain compatibility with a single-user OS/8 installation. Public packs (see 5.11) are normally used for auxiliary disk storage because they allow any number of users to access the pack at the same time.

Before you can access a private pack, you must assign it with the SCALE command ASSIGN (see 3.10, System User's Guide). Once a user has ASSIGNed a disk drive, no other user may ASSIGN the disk successfully. Other users attempting to ASSIGN the pack receive the "DEVICE OWNED BY JOB X" error message. However, any user LOGged INto an account, which has the same project number as the user who ASSIGNed the pack, may access that private pack. The second user simply LOOKS UP the pack without ASSIGNing it. The message "n? FUNCTION FAILURE" is printed if access is denied.

Once you have ASSIGNed the disk, you must associate an ETOS channel with it via the LOOKUP command. The format of the LOOKUP command is contained in Figure 5-46.

Figure 5-46
LOOKUP Command Format

```
!LOOKUP channel=DKn:[,offset]
```

The channel is used to access the device under OS/8 or COS. "n" represents the drive number of the drive containing the private pack if the disk is an RK05. If the device is a System Industries disk, n=1 represents the removable pack in port 0; n=2 represents the fixed pack in port 1; n=3 represents the removable pack in port 1. The offset is not used to access a COS system pack or data pack, or to access the first portion of an OS/8 pack (RKA0 or DSK0). The offset is used to access secondary OS/8 directories (RKB0 or DSK1, DSK2, DSK3) on the private pack.

5.10.1 Accessing a Stand-alone OS/8 Pack

The LOOKUP command with no offset is used to access the "A" side of an OS/8 RK05 pack or the "0" side of an OS/8 System Industries pack. Once the LOOKUP command is performed, standard OS/8 commands can be used to transfer data to and from the private pack. The pack is accessed as "CHNX:", where X represents the

channel used in the LOOKUP command. Figure 5-47 illustrates the transfer of OS/8 files HARM.SV, GOLF.BA and LINEAR.FT from the "A" side of an OS/8 stand-alone pack mounted in RKØ5 drive 1 to an OS/8 storage area in the current account.

Figure 5-47
Sample Access of OS/8 Stand-alone Pack

```
.^VS                          (enter SCALE)
!ASSIGN DK1:                  (assign drive containing pack)
!CLOSE 3                      (insure channel is available)
!LOOKUP 3=DK1:                (associate channel 3 with "A" side)
!CONTINUE                     (return to OS/8)
COPY DSK:<CHN3: HARM.SV,GOLF.BA,LINEAR.FT

FILES COPIED:
HARM.SV
GOLF.BA
LINEAR.FT
.
```

In the above example, to transfer additional files, only a COPY command is required.

To access the "B" side of an OS/8 RKØ5 pack, an offset of 6260 is used in the LOOKUP command. Figure 5-48 illustrates the association of channels 3 and 4 with the "A" and "B" sides of an OS/8 private pack in drive 1.

Figure 5-48
Accessing the "A" and "B" Sides of an OS/8 RKØ5 Pack

```
.^VS
!CLOSE 3
!CLOSE 4
!LOOKUP 3=DK1:
!LOOKUP 4=DK1:,6260
!CONTINUE
```

After the dialogue of Figure 5-48 is followed, the "A" side of the pack in drive 1 can be accessed as CHN3:. The "B" side can be accessed as CHN4:.

The LOOKUP command with no offset is used to access the "Ø" portion of a System Industries pack. To access the "1" portion of the pack, an offset of 6300 is used. To access the "2" portion of the pack, an offset of 14600 is used. to access the "3" portion of the pack, an offset of 23100 is used. Figure 5-49 contains a dialogue which associates the four portions of an OS/8 stand-alone pack with ETOS channels 3 through 6.

Figure 5-49
Accessing the Ø-3 Portions of an OS/8 S.I. Pack

```
.^VS
!CLOSE 3;CLOSE 4;CLOSE 5;CLOSE 6
!!!!LOOKUP 3=DK1:
!LOOKUP 4=DK1:,6300
!LOOKUP 5=DK1:,14600
!LOOKUP 6=DK1:,23100
!CONTINUE
```

After the dialogue of Figure 5-49 is followed, the "Ø" portion of the pack can be accessed as CHN3:. The "1" portion can be accessed as CHN4:. The "2" portion can be accessed as CHN5:. The "3" portion can be accessed as CHN6:.

Figure 5-50 contains a sample of copying the programs HELP.SV and HELP.HL from the OS/8 system storage area to the "2" side of a System Industries OS/8 private pack, mounted in drive 1.

Figure 5-50
Sample Access of S.I. OS/8 Stand-Alone Pack

```
.^VS
!CLOSE 3
!LOOKUP 3=DK1:,14600
!CONTINUE
COPY CHN3:<SYS:HELP.SV,HELP.HL

FILES COPIED:
HELP.SV
HELP.HL
.
```

5.10.2 Accessing a Stand-Alone COS Pack

The LOOKUP command with no offset is used to access a COS system pack or data pack. After the LOOKUP command is issued, the COS pack may be accessed under COS as DKx, where x is the channel used in the LOOKUP command. For example, Figure 5-51 contains a dialogue which transfers a COS source program called PRTOCL and a COS binary program called PGMEND from a COS stand-alone pack mounted in drive 1 to a COS storage area in the current account.

Figure 5-51
Sample Access of Stand-Alone COS Pack

```
.^VS
!CLOSE 3
!LOOKUP 3=DK1:
!CONTINUE
R COSBO
COS MONITOR V3.07GB

.R PIP
PIP 3.07B
OPT - S
IN - PRTOCL,DK3
OUT - PRTOCL

OPT - B
IN - PGMEND,DK3
OUT - PGMEND
OPT - X
COS MONITOR 3.07GB
.
```

Note that in the above example, the stand-alone COS pack was referred to as DK3, even though it was mounted in physical drive 1. Under ETOS, DK numbers refer to ETOS channels which may or may not coincide with the physical drive number. To eliminate confusion, you can associate ETOS channel numbers with their physical drive numbers whenever possible. Channel 1 could be LOOKed UP on physical DK1:, channel 2 could be LOOKed UP on physical DK2: and channel 3 could be LOOKed UP on physical DK3:. However, channel 1 should not be used if you need to return to

OS/8 after running COS. Channel 1 is required for OS/8 scratch blocks, which must be left alone if you want to return to OS/8.

If you wish to transfer data files from a stand-alone COS pack, you must utilize the SYSGET program. SYSGET is the ETOS version of SYSGEN. Read section 5.5 in the System User's Guide for detailed information about SYSGET. Before you can transfer a data file, you must associate a channel with the stand-alone COS pack and a channel with a COS data file area (DATAFI.LES). You must then map the files on the COS pack to logical units via SYSGET. On COS data packs, files are allocated sequentially from the beginning of the disk. On COS system packs, the length of all the files on the disk is added together and subtracted from 406, the total number of segments on a pack. Therefore, if you had three files stored on the system pack with lengths 5, 12 and 20 segments, the files would begin at segment 369. If you wished to access these files on a private COS pack, you must assign a dummy of 369 segments. Figure 5-52 contains a dialogue which transfers two files from a COS data pack in drive 1 and two files from a stand-alone COS system pack in drive 2 to a COS data file are in account [10,20].

Figure 5-52
Sample Transfer of COS Data Files

```
.^VS
!CLOSE 3;CLOSE 4;CLOSE 5
!!!ASSIGN DK1:
!ASSIGN DK2:
!LOOKUP 3=DK1:              (COS data pack)
!LOOKUP 4=DK2:              (COS system pack)
!LOOKUP 5=DK0:[10,20]DATAF1.LES  (COS data file area)
!CONTINUE
R COSBO
COS MONITOR V3.07GB

.ER
.100 DK3, 7;CUSMAS/1
.110 DK3, 2;CUSIDX/2
.120 DK4, 380;DUMMY/3
.130 DK4, 17;INVMAX/4
.140 DK4, 9;INVIDX/5
.150 DK5, 7;CUSMAS/6
.160 DK5, 2;CUSIDX/7
.170 DK5, 17;INVMAS/8
.180 DK5, 9;INVIDX/9
.R SYSGET
DONE
COS MONITOR V3.07GB
.R PIP
PIP 3.07B
OPT - D
IN - CUSMAS/1
OUT - CUSMAS/6

OPT - D
IN - CUSIDX/2
OUT - CUSIDX/7

OPT - D
IN - INVMAS/4
OUT - INVMAS/8

OPT - D
IN - INVIDX/5
OUT - INVIDX/9

OPT - X

COS MONITOR V3.07GB
.
```

In the example of Figure 5-52, it was shown how you can transfer

files by creating a logical unit mapped to a COS data file pack and an equivalent area in a COS data file area. Since each file requires two logical units in this method, you can only transfer seven files at a time. You can then change the SYSGET table, including dummy files to skip past COS system areas or files already transferred.

5.10.3 Creating a Stand-alone OS/8 Pack

For some special purpose software, it may be necessary to run on a single-user OS/8 disk pack. It is possible to create a stand-alone OS/8 pack from a copy of the ETOS distribution pack if you have two RKØ5 drives or a System Industries drive. Use the following procedure to accomplish this creation.

1) Boot a copy of the ETOS distribution pack. Configure the pack for your installation but do not Sysgen it. Start ETOS running.

2) LOGIN to the stand-alone software account ([Ø,11] STANDAL). Mount a blank formatted pack in RKØ5 drive 1 or System Industries port Ø.

3) Enter the dialogue contained in Figure 5-53 in order to create a stand-alone OS/8 V3D disk pack.

Figure 5-53
Creating a Stand-Alone OS/8Disk

```
.^VS
!ASSIGN DK1:
!CLOSE 3;CLOSE 4
!!LOOKUP 3=[Ø,2]OS8.OS8
!LOOKUP 4=DK1:
!CONTINUE
R PIP
*CHN4:<CHN3:/Y/Z          (transfer stand-alone OS/8 Monitor)
*CHN4:DSK:/S/O            (transfer stand-alone OS/8 programs)
*^C
.
```

4) You now have a stand-alone OS/8 pack which can be booted in drive Ø. However, some of the software is incomplete. Account [Ø,11] contains only the stand-alone version, of

programs which must be modified to run under ETOS. Programs which are identical under virtual and stand-alone OS/8 can be copied to CHN4: from SYS:. A list of the programs which must be copied to complete each group is listed in Table 5-6.

Table 5-6
Programs Which Complete Product Groups

Group	Programs Required From SYS
COBOL	COBOL.SV, COBOL1.SV, COBOLA.SV COBOL8.RL, COBOL.HE
EB BASIC	No programs required
EXTENSION KIT	No programs required
FORTRAN IV	FORLIB.RL, RALF.SV
MACREL	No programs required
OS/8	No programs required
QBOL	QBOL.SV, QBOLCM.CB, P2ZZZZ.CB, P3ZZZZ.CB, P4ZZZZ.CB, P5ZZZZ.CB, P6ZZZZ.CB, P7ZZZZ.CB, P8ZZZZ.CB, P9ZZZZ.CB, 6SETUP.SV, 6SORT.SV, 6SHUFL.SV, 6RETRV.SV
SORT/8	No programs required

5.10.4 Creating a Stand-Alone COS Pack

When you bootstrap the ETOS pack, you are running stand-alone OS/8. This OS/8 monitor head can be transferred to create a stand-alone OS/8 pack (see 5.10.3). There is no copy of stand-alone COS on the ETOS pack. Therefore, you cannot create a stand-alone COS pack from the ETOS disk. If you have a stand-alone COS pack, you can transfer programs to it via the procedures of 5.10.2.

5.11 CREATING A PUBLIC PACK

Section 5.10 discussed the access of a stand-alone OS/8 or COS format disk. The drawback in utilizing this type of disk is that access is limited to one user at a time. ETOS also supports public packs, which support multiple users. Public packs can be used to extend the amount of on-line storage. The format of these public packs is similar to the format of the system pack, except many of the system files are not required on non-system packs. When it is initially created, the public pack contains only a Master file directory (0,1) and a library account (0,2), which contains the ETOS file called FREEBLO.CKS. The format of the public pack is illustrated for the RK05 in 1.3.1 and for the S.I. disk in 1.3.2.

To create a public pack, the program DSKINT is used. To run DSKINT, enter R DSKINT<RET>. The program responds with

```
'DSKINT' V2.010
DK TO INITIALIZE?
```

Enter the physical DK number of the disk you wish to initialize. For RK05 sites, this number corresponds to the physical drive. For System Industries sites, a DK number of 1 represents the removable pack in port 0. A DK number of 2 represents the fixed pack in port 1. A DK number of 3 represents the removable pack in port 1. The system responds with

```
DISK INITIALIZED
.
```

The disk is now initialized as a public pack. Note that the pack must be formatted before it can be made a public pack.

A sample initialization of a pack in DK1 is contained in Figure

5-54.

Figure 5-54
Sample Run of DSKINT

```
.R DSKINT
'DSKINT' V2.Ø1Ø
DK TO INITIALIZE? 1
DISK INITIALIZED
.
```

DSKINT sets the PRIV word to 4Ø4Ø, ASSIGNs the disk and looks up channel 7 on the disk. After initialization, DSKINT sets the PRIV word to ØØ4Ø, closes channel 7 and DEASSIGNs the disk.

A public pack can never be used as a system disk. If you attempt to bootstrap a public pack as a system disk, the message "PUBLIC PACK CANNOT BE BOOTED" is printed. Before a public disk can be accessed, it must be MOUNTed (see 6.2.6). This MOUNT process has to be repeated each time ETOS is brought up. A logical time for all MOUNTs is after you initialize the time and date. Figure 5-55 contains the dialogue necessary to MOUNT a public pack in drive 1. The DK number used in the MOUNT is the same one specified to DSKINT.

Figure 5-55
MOUNTing a Public Pack

```
.^VS
!MOUNT DK1:PASSWORD,1
!CONTINUE
```

Public packs have the master file directory stored in block 1. This is the reason you use an offset of 1 in the MOUNT command. System packs have the master file directory stored in block 12ØØ. You must use an offset of 12ØØ to MOUNT a system pack in a non-system drive. The system pack in drive ØØ is automatically MOUNTed by ETOS. You might wish to access the master distribution pack, write locked, by loading it in drive 1 and

copying information from it. Figure 5-56 contains the MOUNT procedure for system packs accessed in a non-system drive.

Figure 5-56
MOUNTing a System Pack

```
.^VS
!MOUNT DK1:PASSWORD,12ØØ
!CONTINUE
```

Once a public pack has been MOUNTed, it is similar to a newly Sysgened system disk. You must create accounts and ETOS files on the disk to utilize it. These accounts and files are created with the ACCNT program. The only difference between creating an account on the system disk and creating it on a public disk is that the ETOS device name is specified as DKx:, where x is 1, 2 or 3, instead of DKØ or SYS.

There are two ways to access a public pack. One method is to create accounts and ETOS files on the public pack which do not reside on the system pack. To access these files, you must use the LOOKup SCALE command or the LOOKUP program (see 5.14). For example, Figure 5-57 contains the access of an OS/8 account storage area in account [5,5Ø] on a public disk MOUNTed on DK1.

Figure 5-57
Accessing a Public Pack Via LOOKUP

```
.^VS
!CLOSE 3
!LOOKUP 3=DK1:[5,5Ø]OS8DISK.DSK
!CONTINUE
DIR CHN3:          (directory of public pack area)
```

The first method of accessing a public pack is acceptable, but requires several commands each time you LOGIN. You can use features of LOGIN and COSBO to make the access of a public pack transparent to all users. The first step in making public pack access easier is to create the desired account number in the

master file directory on the system pack. The reason you must create the account on the system pack even though the storage area is on the public pack is that LOGIN checks only the system pack for valid account numbers and passwords. The second step in this process is to create the desired account in the Master file directory on the public pack. This seems redundant, but an account entry in the Master File Directory is required before you can create ETOS files under the account. The final step in the process is to create the ETOS file under the account on the public pack. The ETOS file is usually an OS/8 storage area, a COS program storage area or a COS data file storage area. Automatic access of these storage areas is discussed in 5.11.1 - 5.11.3. Figure 5-58 contains a dialogue which lists a public pack in DK1 before any accounts or storage areas are created on it.

Figure 5-58
ACCNT List of Initialized Public Pack

```
.^VPRIV 4040
R ACCNT
'ACCNT' V5.53

*OUTPUT<INPUT
*<RET>

ETOS DEVICE NAME?: DK1:
PASSWORD?: PASSWORD

OPTION?: LIST
ACCOUNT?: ALL

00,01 PASSWORD
   00,01 PASSWORD
   00,02 LIBRARY

00,02 LIBRARY
      FREEBLO.CKS <7777>      0040*     32*

FREE BLOCKS:                  14473     6459
```

The List contained in Figure 5-58 is applicable to an RK05 pack.

A list of a System Industries pack is identical except the number of free blocks is 31333 (base 8) or 13Ø19 (base 1Ø).

5.11.1 Creating an OS/8 Storage Area

Once you have created an account in the master file directory of the system disk and the public pack, you can create an OS/8 storage area. Access of this area is transparent to the user. When you LOGIN, LOGIN checks the system disk for the ETOS file OS8DISK.DSK.

If the file is not found, LOGIN checks DK1: for OS8DISK.DSK. if it is not found, DK2: and DK3: are inspected. If the file is not found on any of the disks, the error message "?OS8DISK.DSK NOT FOUND" is printed. However, if the file OS8DISK.DSK is found on any of the disks, LOGIN associates the file with channel 2. The storage area is accessed by referencing OS/8 device CHN2: or DSK: or by omitting the device name entirely. The key element is that the user does not know which disk his storage area is on. Therefore, you can create storage areas on the system pack and all public packs. The disk layout is transparent to the user. Figure 5-59 contains a sample creation of an OS/8 storage area on a public pack MOUNTed in DK1.

Figure 5-59
Sample Creation of OS/8 Storage Area

```
.^VPRIV 4040
R ACCNT
'ACCNT' VERSION 5.53
*OUTPUT<INPUT
*<RET>

ETOS DEVICE NAME?: DK0:
PASSWORD?: PASSWORD

OPTION?: CREATE MFD [17,20] OSTEST 1077
(account now created in system disk's MFD)
OPTION?: DEVICE

ETOS DEVICE NAME?: DK1
PASSWORD?: PASSWORD

OPTION?: CREATE MFD [17,20] OSTEST 1077
(account now created in public pack's MFD)
OPTION?: CREATE [17,20] OS8DISK.DSK <0050> D175 VARIABLE
(OS/8 storage area now created on public pack)
OPTION?: ZERO [17,20]OS8DISK.DSK
(OS/8 storage area now initialized)
OPTION?: EXIT
CHANNELS 5,6,7 CLOSED
.
```

After the dialogue of Figure 5-59 is followed, the dialogue of Figure 5-60 can be used to access the public pack's OS/8 storage are in account [17,20].

Figure 5-60
Automatic Access of Public Pack's OS/8 Storage

```
.^VS
!LOGOUT;F
!LOGIN
ETOS V5B AT 01:29:37 ON TUE 03-APR-79
ACCOUNT? 17,20
PASSWORD? OSTEST
JOB 03 LOGGED IN ON CONSOLE 00
.DIR DSK:/Q

168 FREE BLOCKS
.
```

The user, LOGged INto account [17,20] in the example of Figure 5-60 has an OS/8 storage on the public pack. It is his default DSK: area.

5.11.2 Creating a COS Storage Area

Once you have created an account in the Master file directory of the system disk and the public pack, you can create a COS storage area. Access of this area is transparent to the user. When you LOGIN, the system boots up OS/8. When you enter R COSBO<RET>, COSBO checks the system disk for the file COSDISK.RTS. If the file is not found, COSBO checks DK1 for COSDISK.RTS. If it is not found, DK2 and DK3 are inspected. If the file is not found on any of the disks, the error message "?COSDISK.RTS NOT FOUND" is printed and OS/8 is reBOOTed. However, if the file COSDISK.RTS is found on any of the disks, LOGIN associates the file with channel 0. The COS storage area is accessed by referencing COS device DK0 or by omitting the device name entirely. The key element is that the user does not know which physical disk his storage area is on. Therefore, you can create storage areas on the system pack and on all public packs. The disk layout is transparent to the user. Figure 5-61 contains a sample creation of a COS storage area on the public pack MOUNTed in DK1.

Figure 5-61
Sample Creation of COS Program Storage Area

```
.^VPRIV 4040
R ACCNT
'ACCNT' VERSION 5.53
*OUTPUT<INPUT
*<RET>

ETOS DEVICE NAME?: DK0:
PASSWORD?: PASSWORD

OPTION?: CREATE MFD [50,2] COSUSE 6077
(account now created in system disk's MFD)
OPTION?: DEVICE
```

Figure 5-61 (continued)

```
ETOS DEVICE NAME?: DK1
PASSWORD?: PASSWORD

OPTION?: CREATE MFD [50,2] COSUSE 6077
(account now created in public pack's MFD)
OPTION?: CREATE [50,2]COSDISK.RTS <00,50> D300 VARIABLE
(COS storage area now created on public pack)
OPTIN?: EXIT

CHANNELS 5, 6, 7 CLOSED
.LOGOUT
JOB 03 LOGGED OUT ON CONSOLE 00
10:17.12 P.M. ON WED 07-APR-79
ELAPSED TIME:
1 HOUR(S) 27 MINUTE(S) 48.2 SECONDS
GOOD MORNING
!LOGIN
ETOS V5B AT 01:30:05 ON TUE 03-APR-79
ACCOUNT? 50,2
PASSWORD? COSUSE
JOB 03 LOGGED IN ON CONSOLE 00
.^VS
!LOOKUP 3=DK1:COSDISK.RTS
!CONTINUE
R COSBLD
'COSBLD' V1.012

CHANNEL NUMER? 3
ZERO OUTPUT DIRECTORY? YES
COS SYSTEM BUILT
.COPY CHN3:*.SV<SYS:*.CS

FILES COPIED:
COMP.CS
BREAK.CS
BUILD.CS
CONVEX.CS
CREF.CS
DAFT.CS
DFDIR.CS
LINCHG.CS
OS8BO.CS
PATCH.CS
PIP.CS
PRINT.CS
SORT.CS
SYSGET.CS
UNBRK.CS
UPDATE.CS
.
```

After the dialogue of Figure 5-61 is followed, the dialogue of Figure 5-62 can be used to access the public pack's COS program storage area in account [50,02].

Figure 5-62
Automatic Access of Public Pack's COS Program Storage

```
.^VS
!LOGOUT;F
!LOGIN
ETOS V5B AT 01:35:54 ON TUE 03-APR-79
ACCOUNT? 50,2
PASSWORD? COSUSE
JOB 03 LOGGED IN ON CONSOLE 00
?CANNOT FIND OS8DISK.DSK
.R COSBO
COS MONITOR 3.07GB
.DI,DK0/T

DIRECTORY 03-APR-79

NAME            TYPE        LN          DATE

COMP            V           14          25-JUL-77
BREAK           V           02          31-DEC-79
BUILD           V           09          25-JUL-73
CONVEX          V           10          25-JUL-73
CREF            V           07          25-JUL-73
DAFT            V           10          25-JUL-73
DFDIR           V           03          11-JUN-74
LINCHG          V           03          11-JUN-74
OS8BO           V           02          01-APR-79
PATCH           V           05          25-JUL-73
PIP             V           11          25-JUL-73
PRINT           V           05          25-JUL-73
SORT            V           15          25-JUL-73
SYSGET          V           07          01-APR-79
UNBRK           V           02          01-APR-79
UPDATE          V           09          25-JUL-73
<0086 FREE BLOCKS>
```

5.11.3 Creating a COS Data File Area

Unlike OS/8 storage areas and COS program storage areas, an account which contains a COS data file storage area does not need an MFD entry on the system disk. You require only an MFD entry for an account on the public pack and an ETOS file under the account called DATAF1.LES. The reason that you don't require an MFD entry in the system disk is that you don't LOGIN to a COS data file area. COS data files are accessed only via logical units set up by SYSGET (see 5.5, System User's Guide). Figure 5-63 contains a sample creation of a COS data file area on a public pack MOUNTed in DK1.

Figure 5-63
Sample Creation of COS Data File Area

```
.^VPRIV 4040
R ACCNT
'ACCNT' VERSION 5.53
*OUTPUT<INPUT
*<RET>

ETOS DEVICE NAME?: DK1
PASSWORD?: PASSWORD

OPTION?: CREATE MFD [10,01] COSDATA 0077
(account now created in publi pack's MFD)
OPTION?: CREATE [10,01] DATAF1.LES <0000> D1600 VARIABLE
(COS data file area now created on public pack)
OPTION?: EXIT
CHANNELS 5, 6, 7 CLOSED
.
```

After the dialogue of Figure 5-63 is followed, the dialogue of Figure 5-64 can be used to access the public pack's COS data file area in account [10,01]. This example assumes that you previously created a COS program storage area in account [50,02] on the system disk or the public pack.

Figure 5-64
Access of Public Pack's COS Data File Area

```
.^VS
!LOGOUT;F
!LOGIN
ETOS V5B AT 01:400:02 P.M. ON TUE 03-APR-79
ACCOUNT? 50,2
PASSWORD? COSUSE
JOB 03 LOGGED IN ON CONSOLE 00
?CANNOT FIND OS8DISK.DSK
.R COSBO
COS MONITOR 3.07GD
.ER
.100 DK3, 20, DK1:[10,01]DATAFI.LES; ARMPST/1
.110 DK3, 5; ARINDX/2
.120 DK3, 75; ARTOGL/3
.R SYSGET
'SYSGET' V2.010
DONE
.R PIP
PIP V1.012
OPT - D
IN - ARTOGL/3
OUT - /K
OPT - X

COS MONITOR 3.07GB
.
```

5.12 MOVING FILES BETWEEN VIRTUAL OS/8 AND ETOS

There are several programs or files which must exist as ETOS files rather than OS/8 or COS files. An example of such a program is SYSTAT, which performs privileged operations but can be run by non-privileged users. You may also write programs which you wish to run as ETOS files. To accomplish this, create the file in an OS/8 storage area and transfer the file from OS/8 to ETOS using the OSETOS program. If the file will be used as a text file (e.g., LOGIN message), create the file with EDIT or TECO. If the file will be run as a program (e.g., SYSTAT), create the assembler program with EDIT or TECO, assemble it with PAL8, load it with ABSLDR and save it as a core image file.

The general format of the OSETOS program is only used if you write a special purpose task which you wish to transfer into an ETOS file. System management functions (e.g., changing the LOGIN message) have special arguments which faciliate loading of the file. These special arguments are discussed in 5.12.1-5.12.5.

To run OSETOS, LOGIN to a privileged account and enter R OSETOS<RET>. The system responds with

```
'OSETOS' V1.Ø32
```

Enter the name of the OS/8 file which you wish to transfer to an ETOS file. The system responds with

```
TO CHANNEL?
```

At this point, you must exit to SCALE and use the ENTER command (see 7.2.3, System User's Guide) to create an ETOS file of the proper length. If the extension of the OS/8 file is ".SV", OSETOS assumes it is a core image file and strips off the core control block. Therefore, the ETOS file is always the same size as the OS/8 file or one block less. After the ENTER command has been specified, CONTinue with OS/8 processing and enter the channel number used in the ENTER. The system responds with

```
FILE LOADED
*
```

You may enter another file to transfer or enter EXIT to terminate execution of the program.

A sample run of OSETOS, which transfers a 17 block core image file called USER.SV from your OS/8 storage area to an ETOS file called USER.SAV in your account is contained in Figure 5-65.

Figure 5-65
Sample Run of OSETOS

```
.R OSETOS
'OSETOS' V1.Ø32
*USER.SV
TO CHANNEL? ^VS
!CLOSE 3
!ENTER 3=USER.SAV,<4152>,2Ø
!CONTINUE
3
FILE LOADED
*^C
.
```

OSETOS is often used to allow a privileged program to be run by non-privileged users. See 6.7 for more information about temporary privilege.

OSETOS sets your PRIV word to 4040. If the general form of the program is used, OSETOS does not affect the status of the ENTERed channel. If the special form of the program is used (see 5.12.1 - 5.12.5), OSETOS closes channel 7, deletes the old ETOS file and LOOKs UP the new ETOS file on channel 7. When the program is exited, the PRIV word is set to 0040 and channel 7 is closed if it was open.

A list of all possible OSETOS error messages follows.

?BAD NUMBER

A channel number was entered, which was not in the range 2 through 7.

?X DELETE ERROR

A deletion error occurred while attempting to delete an ETOS file which is being replaced. "X" corresponds to the errors in the DELETE CHANIO instruction (see 7.4.9 System User's Guide). This

error usually indicates that the file is being accessed by another user.

FILE NOT FOUND

The OS/8 file which was specified does not exist.

?PROTECTION VIOLATION

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

?X WRITE ERROR

A write error occurred while attempting to write information to the ETOS file. "X" corresponds to the errors in the WRITEW CHANIO instruction (see 7.4.3, System User's Guide). This error normally indicates a corrupt ETOS file. Delete it and recreate it to solve the problem.

5.12.1 Loading SYSTAT

The SYSTAT program provides you with a system status of any point in time. See 4.5, System User's Guide for complete documentation on this program. The program SYSTAT.SV resides on virtual OS/8 SYS:. The OSETOS program is used to transfer SYSTAT.SV from virtual OS/8 to the ETOS file SYSTAT.SAV in the account [ØØ,Ø2]. The SYS SCALE command and the SYS CCL command require that this transfer be completed before you can obtain a system status. Follow the dialogue contained in Figure 5-66 to load SYSTAT into an ETOS file.

Figure 5-66
Loading SYSTAT

```
.R OSETOS
'OSETOS' V1.Ø32
*SYS:SYSTAT.SV
FILE LOADED
*EXIT
.
```

The reason that SYSTAT must be run as an ETOS file is that it is a privileged program able to be run by non-privileged users. However, there is no danger of a non-privileged user obtaining permanent privilege by running SYSTAT. See 6.7 for more information about temporary privilege.

The source version of SYSTAT is contained in the file SYSTAT.PA which is stored on the distribution pack in account [Ø0,Ø7]. You can modify the SYSTAT program and transfer a modified core image file to an ETOS file with OSETOS. You must name your core image file SYSTAT.SV in order to key OSETOS to transfer it to the proper ETOS file.

5.12.2 Loading DKCOPY

The DKCOPY program allows you to copy a disk (system or non-system) while ETOS is running. See 6.6 for complete information on this program. The program DKCOPY.SV resides on virtual OS/8 SYS:. The OSETOS program is used to transfer DKCOPY.SV from virtual OS/8 to the ETOS file DKCOPY.SAV in the account [Ø0,Ø2]. Follow the dialogue contained in Figure 5-67 to load DKCOPY into an ETOS file.

Figure 5-67
Loading DKCOPY

```
.R OSETOS
'OSETOS' V1.Ø32
*SYS:DKCOPY.SV
FILE LOADED
*EXIT
.
```

The reason DKCOPY must be run as an ETOS file is for security purposes. By maintaining DKCOPY in the ETOS directory, it is separate from other files which are regularly accessed by all users.

The source version of DKCOPY is contained in the file DKCOPY.PA, which is stored on the distribution pack in account [Ø0,Ø7]. You can modify the DKCOPY program and transfer a modified core image file to an ETOS file with OSETOS you must name your core image file DKCOPY.SV in order to key OSETOS to transfer it to the proper ETOS file.

5.12.3 Loading LOGIN

The ETOS file LOGIN.SAV which resides in account [Ø,2] handles all LOGIN and LOGOUT processing. Unlike SYSTAT and DKCOPY, this file automatically exists as an ETOS file after performing a Sysgen. the only reason for loading a new copy of LOGIN is if you wish to modify LOGIN/LOGOUT processing. The source to LOGIN is contained in the file LOGIN.PA, which is stored on the distribution pack in account [Ø0,Ø7]. An OS/8 LOGIN image file is contained in virtual OS/8 SYS. You can modify LOGIN.PA and create a core image file, which can be transferred to an ETOS file with OSETOS. You must name your core image file LOGIN.SV in order to key OSETOS to transfer it to the proper ETOS file. Follow the dialogue contained in Figure 5-68 to load a new copy of LOGIN.

Figure 5-68
Loading LOGIN

```
.R OSETOS
'OSETOS' V1.Ø32
*LOGIN.SV or SYS:LOGIN.SV
FILE LOADED
*EXIT
.
```

5.12.4 Loading an Installation File

The SYSTAT program (see 4.5, System User's Guide) prints a header indicating the name of the installation. After a Sysgen, the name is printed as "QUODATA CORPORATION". You can change this name to the title of your company or school. The installation name is taken from an ETOS file called INSTALL.TXT which resides in [Ø0,Ø2]. OSETOS is used to transfer an OS/8 text file, which must be called INSTAL.TX, to the ETOS file. INSTAL.TX may be created with EDIT or TECO. A sample load of an installation file is contained in Figure 5-69.

Figure 5-69
Loading an Installation File

```
.CREATE INSTAL.TX        (create OS/8 text file)
#A                       (append)
ABC COMPANY              (name of installation)
<CTRL/L>                 (exit insert mode)
#E                       (enter text file)
.R OSETOS
'OSETOS' V1.Ø32
*INSTAL.TX
FILE LOADED
*EXIT
.
```

After the dialogue of Figure 5-69 is followed, SYSTAT prints "ETOS V5B FOR ABC COMPANY ON ..." as the header line.

5.12.5 Loading a LOGIN Message

When a user enters a valid account number and password, LOGIN performs the necessary initialization. LOGIN then checks to see if an ETOS file LOGMESG.TXT resides in account [0,2]. If the file exists, the contents are printed on the terminal. OSETOS is used to transfer an OS/8 text file, which must be called LOGMES.TX, to the ETOS file. This LOGIN message contained in the text file may be any length and consist of any number of lines. Figure 5-70 contains a sample dialogue which loads a LOGIN message.

Figure 5-70
Loading a System-Wide LOGIN Message

```
CREATE LOGMES.TX                    (create OS/8 text file)
#A                                  (append)
WELCOME TO QUODATA'S ETOS SYSTEM    (message, line 1)
TYPE HELP TO OBTAIN ASSISTANCE      (message, line2)
<CTRL/L>                            (exist append mode)
#E                                  (enter text file)
.R OSETOS
'OSETOS' V1.032
*LOGMES.TX
TO ACCOUNT? 0,2
FILE LOADED
.
```

After the system-wide LOGIN message is displayed, LOGIN checks the user account to see if it contains an ETOS file called LOGMESG.TXT. If such a file exists, the contents are printed on the terminal. Therefore, the system manager can set up messages that will only be displayed when selected users LOGIN to the system. Figure 5-71 contains a dialogue which creates an additional LOGIN message for account [17,20].

Figure 5-71
Loading a User Account LOGIN Message

```
.CREATE LOGMES.TX
#A
THIS MESSAGE IS ONLY FOR [17,20]
<CTRL/L>
#E
.R OSETOS
'OSETOS' V1.Ø32
*LOGMES.TX
FILE LOADED
.
```

If you want to change the LOGIN message, create a new LOGMES.TX file and transfer it with OSETOS. If you want to eliminate the system-side LOGIN message or a user account LOGIn message, transfer a blank file. Figure 5-72 contains the dialogue which can be used to eliminate the system-wide LOGIN message.

Figure 5-72
Eliminating a LOGIN Message

```
.CREATE LOGMES.TX
#A
<CTRL/L>
#E
.R OSETOS
'OSETOS' V1.Ø32
*LOGMES.TX
FILE LOADED
.
```

5.13 MOVING FILES BETWEEN ACCOUNTS

When you bootstrap the ETOS disk (see 2.2 or 2.3), you are running single-user or real OS/8. This copy of OS/8 is unmodified. After you start ETOS and LOGIn, you automatically have access to a time sharing or virtual version of an OS/8 system. This OS/8 system area, which is accessed as SYS, contains all programs (e.g., PIP.SV, DIRECT.SV) which must be accessed by all OS/8 users. You can add programs of general

interest to this area. The main advantages of this virtual OS/8 system is that it is automatically accessible by all users, thus saving disk space, and it is write protected for non-privileged users, thus maintaining security. LOGIN also checks to see if an OS/8 storage area (OS8DISK.DSK) exists for the account on the system disk or any of the public packs. If the storage area exists, it is accessible as DSK: or CHN2:. This area is a private storage area, which is not automatically accessible to any other user. Under OS/8 (real, virtual SYS or storage area), programs and data files can be stored in the same area, since both are listed in the OS/8 directory.

After you LOGIN, entering R COSBO<RET> causes the system to check if a COS program storage area (COSDISK.DSK) exists for the account on the system disk or any of the public packs. If the storage area exists, it is accessible as DKØ. This area is a private storage area, which is not automatically accessible to any other user. Unlike OS/8, there is no common system area with COS programs (e.g., PIP, DTDIR). Each user has his own copy of the programs he wishes to run.

Under COS, programs and data files are not normally stored in the same storage area. Since data files are allocated from the end of the device in program areas, your flexibility in changing the layout of the storage area is greatly reduced. When data files are stored in their own storage area, files are allocated from the beginning of the area. Adding data files to the area is simplified using this method. Since data files do not appear in a COS directory, they can only be accessed as logical units. SYSGET is used to associate logical units with COS data file storage areas (DATAF1.LES).

This section documents how to move programs and files between the five types of storage areas used under COS. These types are listed in Table 5-7.

Table 5-7
Types of Storage Areas

Accessed by Operating System	Documentation Title	Description
OS/8	Real OS/8 or stand-alone OS/8 or single-user OS/8	Unmodified version of DEC's OS/8 operating system which is initiated by bootstrapping the physical machine. Only the console terminal is active. ETOS is initiated from this area. This area is accessed as SYS: under single-user mode.
OS/8	Virtual OS/8 System	Timesharing version of DEC's OS/8 operating system, which is automatically accessible as SYS: after LOGIN. This write-protected area contains all programs which need to be accessed by all users.
OS/8	OS/8 Storage Area or OS8DISK.DSK	Private storage area for OS/8 programs and files, which is accessed as CHN2: or DSK:. This area contains files which are unique to a particular ETOS user. A private OS/8 storage area may not exist for all accounts.
COS	COS Program Storage Area or COSDISK.RTS	Private storage area for COS programs, which is accessed as DKØ. This area contains all files which a user requires for processing. A private COS storage area may not exist for all accounts.
COS	COS Data File Area or DATAFI.LES	Data file storage area, which may be accessed by more than one user simultaneously. Data file areas may only be accessed as logical units, which are set up with SYSGET.

Table 5-57 (continued

Note: The documentation in all of the subsections of 5.13 assume that you are LOGged INtp a privileged account, which contains a COS storage area if applicable. These examples are valid regardless of the protection code on the storage areas. If the storage area's protection code (see 5.2.2) allows access by other user's, non-privilged users can execute these examples. Non-privileged users cannot use the PRIV command or the WE command, but these commands are only necessary to override protection codes.

5.13.1 Moving Files Between OS/8 Storage Areas

It may be desirable to move OS/8 programs and files between two private user storage areas. To accomplish this transfer, you must LOOKUP each area (OS8DISK.DSK) on an ETOS channel and utilize OS/8 commands to copy the files between the channels. Figure 5-73 contains an example of moving a program called PROG.SV from a private OS/8 storage area on the system pack, account [7,17], to a private OS/8 storage area on a public pack in DK1, account [15,00]. The example also transfers a data file called DATA.DA from [15,00] to [7,17].

Figure 5-73
Moving Files Between OS/8 Storage Areas

```
.^VS
!PRIV 4040          (not needed if storage area protection codes
                     allow non-privileged access)
!CLOSE 3
!CLOSE 4
!LOOKUP 3=DK0:[7,17]OS8DISK.DSK
!LOOKUP 4=DK1:[15,0]OS8DISK.DSK
!WE 3               (not needed if non-privileged access allowed)
!WE 4               (not needed if non-privileged access allowed)
!CONTINUE
COPY CHN4:<CHN3:PROG.SV (copy from channel 3 to channel 4)
FILES COPIED:
PROG.SV

.COPY CHN3:<CHN4:DATA.DA (copy from channel 4 to channel 3)
FILES COPIED:
DATA.DA
.
```

5.13.2 Moving Programs between COS Storage Areas

If you have a program stored in a COS storage area which you wish to run in another COS program, you must transfer the program to the desired area. To accomplish this transfer, you must LOOKUP each area (COSDISK.RTS) on an ETOS channel and use COS commands to copy programs between the channels. Figure 5-74 contains an example of moving a save image program called COMP from a COS storage area on the system pack, account [7,17], to a COS storage area on a public pack in DK1, account [15,00]. The example also transfers a batch file called STREAM from [15,00] to [7,17].

Figure 5-74
Moving Programs Between COS Storage Areas

```
.R COSBO
COS MONITOR 3.07GB

.^VS
!PRIV 4040      (not needed if non-privileged access allowed)
!CLOSE 3
!CLOSE 4
!LOOKUP 3=DK0:[7,17]COSDISK.RTS
!LOOKUP 4= DK1:[15,0]COSDISK.RTS
!WE 3           (not needed if non-privileged access allowed)
!WE 4           (not needed if non-privileged access allowed)
!CONTINUE
R PIP
PIP V3.07
OPT - V
IN - COMP,DK3
OUT - COMP, DK4 (transfer from channel 3 to channel 4)

OPT - S
IN - STREAM, DK4
OUT - STREAM DK3

OPT - X
COS MONITOR V3.07GB
.
```

5.13.3 Moving Files Between COS Data File Areas

If you change the layout of a data file area, you may wish to create another data file area and copy files from one data file area to the other. To accomplish this transfer, you must LOOKUP each area (DATAFI.LES) on an ETOS channel, use SYSGET to associate each data file with a logical unit, and use COS commands to copy files between the channels. Figure 5-75 contains an example of moving a data file called COSMAS from a COS data file area called CUSMAS on the system pack, account [7,17], relative segments Ø-3, to a COS data file area on a public pack in DK1, account [15,ØØ], relative segments 1Ø-12. This example also transfers a data file called APTOGL from [15,ØØ], relative segments 5-9, to [7,17], relative segments 4-8.

Figure 5-75
Moving Files Between COS Data File Areas

```
.R COSBO
COS MONITOR 3.Ø7GD

.^VS
!PRIV 4Ø4Ø      (not needed if non-privileged access allowed)
!CLOSE 3
!CLOSE 4
!LOOKUP 3=DKØ:[7,17]DATAF1.LES
!LOOKUP 4=DK1: [15,Ø]DATAF1.LES
!WE 3           (not needed if non-privileged access allowed)
!WE 4           (not needed if non-privileged access allowed)
!CONTINUE
ER
.1ØØ DK3,4; CUSMAS/1
.11Ø DK3,5; APTOGL/2
.12Ø DK4,5; DUMMY/3
.13Ø DK4,5; APTOGL/4
.14Ø DK4,4; CUSMAS/5
.R SYSGET
'SYSGET' V2.Ø1Ø
DONE
.R PIP
PIP V3.Ø7B
OPT - D
IN - CUSMAS/1
OUT - CUSMAS/5 (transfer from channel 4 to channel 3)
MORE? N

OPT - X
COS MONITOR 3.Ø7GB
.
```

5.13.4 Moving Programs Between OS/8 and COS Storage Areas

You may have programs which you might want to transfer from OS/8 to COS or from COS to OS/8. This situation occurs if you are converting a DIBOL program to QBOL or you are converting a QBOL program to DIBOL. There are two alternative methods which can be used to accomplish this transfer. With either method, you must LOOKUP the OS/8 storage area (OS8DISK.DSK) and the COS storage area (COSDISK.RTS) on another channel. Using the first method, you can utilize OS/8 commands to copy programs between the channels. Using the second method, you can enter COS and utilize

COS commands to copy programs between the channels. Either method is acceptable.

When transferring programs between OS/8 and COS, you must be aware of the fact that the directory structure is identical, but each operating system looks at the program names differently. OS/8 has a one to six character program name and a one to two character extension. COS has a one to six character program name and a one character program type. When you are under OS/8, utilize the name and extension. When you are under COS, utilize the name and type. Table 5-8 contains the equivalencies between OS/8 extensions and COS program types.

Table 5-8
OS/8 and COS Naming Conventions

OS/8 Extension	Equivalent COS Program Type	Description
SV	V	Save image file
AS	S	ASCII source file
DB	B	Binary compiled file
DF	F	Forms specification file

Figure 5-76 contains an example illustrating both methods of program transfer. A batch file called CHECKS is transferred from an OS/8 storage on the system disk, account [7,17] to a COS storage area on the public disk in DK1, account [15,Ø]. A binary program called EMAINT is then transferred from [15,Ø] to [7,17], using COS commands.

Figure 5-76
Moving Programs Between OS/8 and COS Storage Areas

```
.^VS
!PRIV 4040     (not needed if non-privileged access allowed)
!CLOSE 3
!CLOSE 4
!LOOKUP 3=DK0:[7,17]OS8DISK.DSK
!LOOKUP 4=DK1:[15,0]COSDISK.RTS
!CONTINUE
COPY CHN4:<CHN3:CHECKS.AS (copy from channel 3 to channel 4)
CHECKS.AS
.R COSBO
COS MONITOR 3.07GB

.R PIP
PIP 3.07
OPT - B
IN - EMAINT,DK4
OUT - EMAINT,DK3 (copy from channel 4 to channel 3)

OPT - X
COS MONITOR 3.07GB
.
```

5.13.5 Moving Files Between OS/8 Storage and COS Data File Areas

You may desire to move data files between OS/8 and COS. This situation occurs if you wish to back up a file on a device which is not supported under COS (e.g., DECtape). You can transfer the file from COS to OS/8 and then use OS/8 to back up the file. The process can be reversed to restore the file.

To accomplish the transfer of data files, you must lookup an OS/8 storage area (OS8DISK.DSK) on a channel, LOOKUP a COS data file area (DATAFI.LES) on another channel, enter COS, use SYSGET to associate the file with a logical unit and use CONVEX (see 3.7, COS system User's Guide) to complete the transfer. Figure 5-77 contains an example of copying a data file called DATA.DA from an OS/8 storage area on the system disk, account [7,17] to a COS data file area on the public pack in DK1, account [15,0], relative segments 0-9. The example also transfers a data file

called INVMAS from [15,0], relative segments 10-12, to [7,17].

Figure 5-77
Moving Files Between OS/8 Storage and COS Data File Areas

```
.^VS
!PRIV 4040      (not needed if non-privileged access allowed)
!CLOSE 3
!CLOSE 4
!LOOKUP 3=DK0:[7,17]OS8DISK.DSK
!LOOKUP 4=DK1:[15,0]COSDISK.RTS
!CONTINUE
R COSBO
COS MONITOR 3.07GB
.ER
.100 DK4,10; DATA/1
.110 DK4,3; INVMAS/2
.R SYSGET
'SYSGET' V2.010
DONE
.R CONVEX
VERSION 3.07

INPUT MODE - A
FILE NAME - DATA.DA, DK3

OUTPUT MODE - D
FILE NAME - DATA/1 (transfer from channel 3 to channel 4)

INPUT MODE - D
FILE NAME - INVMAS/2

OUTPUT MODE - A
FILE NAME - INVMAS.DA, DK3 (transfer from channel 4 to
            channel 3)

INPUT MODE - ^C

COS MONITOR 3.07GD
.
```

5.13.6 Moving Files Between Real and Virtual OS/8

If you have a device on your system which is not supported under ETOS (e.g., floating point processor), you can develop programs using the device under ETOS and run them under single-user (real). If you have programs stored on a device which is not

supported under ETOS (e.g., cassette), you can transfer them to real OS/8 under single-user mode and run them under ETOS. For either of these situations, you need to be able to transfer files between real OS/8 and virtual OS/8. To accomplish this transfer you must lookup the stand-alone OS/8 area (OS8.OS8) on an ETOS channel and a private OS/8 storage area on another channel and use OS/8 commands to copy the files between the channels. Figure 5-78 contains an example of moving a program called CSDEMO.BA from real OS/8 to a private OS/8 storage area on the system disk, account [15,00]. The example also transfers a program called FPPUSE.FT from [15,00] to real OS/8.

Figure 5-78
Moving Files Between Real and Virtual OS/8

```
.^VS
!PRIV 4040      (not needed if non-privileged access allowed)
!CLOSE 3
!CLOSE 4
!LOOKUP 3=DK0:[0,2]OS8.OS8
!LOOKUP 4=DK0:[15,00]OS8DISK.DSK
!WE 3           (not needed if non-privileged access allowed)
!WE 4           (not needed if non-privileged access allowed)
!CONTINUE
COPY CHN4:<CHN3:CSDEMO.BA (from channel 3 to channel 4)
FILES COPIED:
CSDEMO.BA
.COPY CHN3:<CHN4:FPPUSE.FT (from channel 4 to channel 3)
FILES COPIED:
FPPUSE.FT
.
```

5.13.7 Moving Files Between OS/8 Virtual System Storage and OS/8 Storage Areas

Virtual OS/8 SYS is accessible to all users. It is write-protected for security reasons. All OS/8 CUSPS (e.g., DIRECT.SV, PIP.SV) and ETOS CUSPS (e.g., DSKINT.SV, SYSTAT.SV) are stored in this area. If you have a program which you wish to be accessible to all users, you may override the write protection and copy it from a private OS/8 storage area to the system device. If you wish to run a modified copy of a system program, you can copy the

program into an OS/8 storage area and run it from that area. Therefore, you must be able to transfer files between OS/8 system storage and OS/8 private storage areas. To accomplish this transfer, you must LOOKUP the private OS/8 storage area (OS8DISK.DSK) on a channel and use OS/8 commands to copy files between the channel and SYS. Figure 5-79 contains an example of copying a program called GENERA.BA from OS/8 SYS to a private OS/8 storage area on a public pack in DK1, account [15,Ø]. The example also transfers a program called PIP.SV from [15,Ø] to OS/8 SYS:.

Figure 5-79
Moving Files Between OS/8 Virtual System Storage and OS/8 Storage Areas

```
.^VS
!PRIV 4Ø4Ø      (always required, if writing on SYS)
!WE Ø           (always required, if writing on SYS)
!LOOKUP 3=DK1:[15,Ø]COSDISK.RTS
!WE 3           (not required if non-privileged access
                allowed)
!CONTINUE
COPY CHN3:<SYS:GENERA.BA (transfer from SYS to channel 3)
FILES COPIED:
GENERA.BA
.COPY SYS:<CHN3:PIP.SV (transfer from channel 3 to SYS)
FILES COPIED:
PIP.SV
.
```

5.14 FACILITATING ETOS FILE ACCESS

After reading section 5.13, it is obvious that the LOOKUP SCALE command is utilized frequently. Any time a user wishes to access any device other than SYS, DSK or TTY, LOOKUP is used. It is not desirable for novice users to enter SCALE. If a mistake is made, they don't know how to recover from it. Therefore, a LOOKUP program is distributed, which eliminates the need for a user to enter SCALE for LOOKUP commands. To run this program, enter R LOOKUP<RET>. The system responds with

```
'LOOKUP' V1.Ø1Ø
1)DK NUMBER?
```

Enter the DK number of the disk containing the ETOS file you wish to access. Ø represents the system disk. On an RKØ5 system 1-3 represents the physical drive number of the public disk. On a System Industries system, 1 represents the removable pack in port Ø; 2 represents the fixed pack in port 1; 3 represents the removable pack in port 1. The system responds with

```
2)ACCOUNT NUMBER?
```

Enter the number of the account which contains the ETOS file you wish to accesscount numbers should be entered without brackets, with the project number and programmer number separated by a comma. The system responds with

```
3)ETOS FILENAME?
```

Enter the name of the ETOS file you wish to access. this name is usually OS8DISK.DSK, COSDISK.RTS or DATAFI.LES. The system responds with

```
4)CHANNEL?
```

Enter the name of the channel which you wish to associate the specified ETOS file with. Any file currently LOOKed UP on that channel will be CLOSEd. The system responds with

```
5)WRITE ENABLE?
```

Enter Y if you wish to override protection codes and write enable the storage area.

Enter N if you wish to allow the ETOS file protection code (see

5.2.2) to control write access. If you run this program in a non-privileged account, this question is not asked. If the LOOKUP is successful, the system responds with

```
FILE LOOKED UP
.
```

The ETOS file is now accessible as CHNX under OS/8 and DKX under COS, where X is the specified channel number. Figure 5-80 contains an example of looking up a private OS/8 storage area (OS8DISK.DSK) on a public pack in DK1, account [7,17].

Figure 5-80
Sample Run of LOOKUP Program

```
.R LOOKUP
'LOOKUP' V1.010
1)DK NUMBER? 1
2)ACCOUNT NUMBER? 7,17
3)ETOS FILENAME? OS8DISK.DSK
4)CHANNEL? 3
5)WRITE ENABLE? Y
FILE LOOKED UP
.
```

Following is a list and explanation of all possible error messages printed by LOOKUP.SV.

```
?X LOOKUP ERROR
```

A lookup error occurred while attempting to lookup the specified ETOS file on the specified channel. X corresponds to the error numbers in the lookup CHANIO function (see 7.4.6, System User's Guide).

Sections 5.14.1 and 5.14.2 contain extended options which can be used to tailor the LOOKUP program to your site. These options are not required. The LOOKUP program may be used as described in 5.14, with no modifications.

5.14.1 Setting LOOKUP Defaults

In the standard operation of LOOKUP, each question requires an answer. If you enter <RET> only, the question is asked again. It is possible to set up defaults for each question. If this is done, entering <RET> only, causes the default to be used. Locations 21-31 in SYS:LOOKUP.SV contain the defaults for all questions. These locations can be modified with FUTIL. Table 5-9 contains a layout of the default locations.

Table 5-9
LOOOKUP Default Locations

Locations	Default	Format
21	DK Number	Octal number in the range Ø to 3.
22	Account Number	Octal number in the range ØØØØ to 7777. The first two octal digits represent the project number. the final two octal digits represent the programmer number.
23-77	ETOS Filename	Ten six-bit ASCII characters wich make up the ETOS file name. The first seven characters make up the filename and the last three characters make up the file extension. File names of less than seven characters or extensions of less must be padded on the right with nulls. No periods are included with the extension.
3Ø	Channel	Octal number in the range Ø to 7.
31	Write Enable	Eight bit ASCII code for "Y" (331) or "N" (316).

Figure 5-81 contains a sample setting of defaults and a subsequent run of the LOOKUP program, using the defaults. The default DK number is 1. The default account number is [2,37]. The default channel is 3. The default state of write enabling is

"Y". The ETOS file OS8DISk.DSK is LOOKed UP on channel 4 on DK0 in account [2,37] by the dialogue specified in the example.

Figure 5-81
Sample Use of LOOKUP Defaults

```
.^VPRIV 4040
^VWE 0
R FUTIL
FILE LOOKUP
LOOKUP.SV  ssss-eeee 0005 (0005)  b.111 01-APR-79
SET MODE SAVE
21/ 0000 1 <LF>                   (default DK)
0000.0022\ 0000  0237<LF>         (default account)
0000.0023\ 0000  1723<LF>         (default ETOS File /OS)
0000.00024\ 0000  7004<LF>        (8D)
0000.00025\ 0000  1123<LF>        (IS)
0000.00026\ 0000  1304<LF>        (KD)
0000.00027\ 0000  2313<LF>        (SK)
0000.00030\ 0000  0003<LF>        (default channel)
0000.00031\ 0000  0331<RET>       (default write enable status)
WRITE
EXIT
.R LOOKUP
'LOOKUP' V1.010
1)DK NUMBER? 0<RET>               (default 1 overrided)
2)ACCOUNT NUMBER? <RET>           (default 02,37 used)
3)ETOS FILENAME? <RET>            (default OS8DISK.DSK used)
4)CHANNEL? 4<RET>                 (default 3 overrided)
FILE LOOKED UP
.
```

Figure 5-81 illustrates how you may use any or all of the defaults. If you wish to disable a default, set the location or locations corresponding to that question to 0000.

5.14.2 Inhibiting LOOKUP questions

When using the LOOKUP program, there may be some questions which always have the same answer. For example, if you are only running OS/8, you might desire that the ETOS filename always be specified as OS8DISK.DSK. Locations 32 - 36 contain a table, which controls which questions are asked. If the entry for a question contains a "1", the question is asked and the default

answer is used only if the user enters <RET>. If the entry for a question contains a "Ø", the question is not asked and the default answer (see 5.14.1) is used. FUTIL is used to set the locations in LOOKUP.SV. Table 5-1Ø contains a list of which locations correspond to which question numbers.

Table 5-1Ø
LOOKUP Question Locations

Location	Question
32	DK NUMBER
33	ACCOUNT NUMBER
34	ETOS FILENAME
35	CHANNEL
36	WRITE ENABLE

Figure 5-82 contains a sample disabling of questions 1, 3 and 5 and the subsequent use of the LOOKUP program. This dialogue assumes that the dialogue of Figure 5-81 was used to set defaults. The ETOS file OS8DISK.DSK residing on DK1 in account [Ø1,5Ø] is looked up on channel 3 with this dialogue.

Figure 5-82
Sample Use of Inhibiting LOOKUP Questions

```
.^VPRIV 4Ø4Ø
^VWE Ø
R FUTIL
FILE LOOKUP
LOOKUP.SV  ssss-eeee ØØ32 (ØØ2Ø)  b.111  Ø1-APR-79
SET MODE SAVE
31/ ØØØ1 Ø<LF>          (disable question 1)
ØØØØ.ØØØ33\ ØØØ1 <LF>   (leave question 2 enabled)
ØØØØ.ØØØ34\ ØØØ1 Ø<LF)  (disable question 3)
ØØØØ.ØØØ35\ ØØØ1 <LF>   (leave question 4 enabled)
ØØØØ.ØØØ36\ ØØØ1 Ø<RET>                  (diable question 5)
WRITE
EXIT
.R LOOKUP
'LOOKUP' V1.Ø1Ø
2)ACCOUNT NUMBER? Ø1,5Ø
Z)CHANNEL? <RET>
FILE LOOKED UP
.
```

5.15 ETOS SYSGEN ACCOUNTS

The Sysgen option of ETOS (see 4.2) creates an initialized ETOS file structure. When you execute this option on a copy of the distribution pack, 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 do not want this space wasted for daily operations. If you want to access this information, you can create a copy of the original distribution pack and operate it without performing a Sysgen. You could also place this copy in a non-system drive and access the information while running from your operation pack.

The next section contains a list of the informational files and accounts. This section discusses only the accounts and files created by the Sysgen option of ETOS. Only the system accounts [0,1], [0,2] and [0,3] are created by this option. A list of the files created under these accounts is contained in the following subsections.

5.15.1 Master File Directory (Account [00,01]

Account [00,01] is the Master File Directory (MFD). The password for account [00,01] is "PASSWORD". You cannot LOGIN to this account. It contains a list of each account on the system and the four account attributes (account number, password, protection and account creation date). Table 5-11 contains a list of the contents of account [0,1] on a newly Sysgened disk. There is no creation date for these accounts.

Table 5-11
MFD Contents

Account Number	Password	Protection Code
[00,01]	PASSWORD	7777
[00,02]	LIBRARY	7777
[00,03]	OPERATOR	7777

If you wish to create an account storage area for OS/8 or COS, you must first enter the account in the Master file directory. After the account is created, a storage area in the account's User File Directory (UFD) can be created. Each account entered in the Master file directory has a one block minimum overhead for its UFD.

5.15.2 Library (Account [00,02])

Account [00,02] is the library account. The password for account [00,02] is "LIBRARY". You cannot LOGIN to this account. It contains ETOS files which are of system-wide use (e.g., virtual OS/8, LOGIN program). Table 5-12 contains a list of the contents of account [0,2] on a newly Sysgened RK05 disk.

Table 5-12
RK05 Library Contents

ETOS filename	Length (Decimal)	Length (Octal)	Protection Code	Fixed or Variable
FREEBLO.CKS	1	1	7777	VARIABLE
IMAGE.RK5	6496	14540	0077	FIXED
IMAGE.RKA	3248	6260	0077	FIXED
IMAGE.RKB	3248	6260	0077	FIXED
LOGIN.SAV	9	11	5152	FIXED
OS8.OS8	640	1200	0052	FIXED
OS8.RTS	1728	3300	1152	VARIABLE
SWAPTRA.CKS	VARIES	VARIES	7777	FIXED

Table 5-13 contains a list of the contents of account [0,2] on a newly Sysgened System Industries disk.

Table 5-13
System Industries Library Contents

ETOS filename	Length (Decimal)	Length (Octal)	Protection Code	Fixed or Variable
FREEBLO.CKS	1	1	7777	VARIABLE
IMAGE.RK5	13056	31400	0077	FIXED
IMAGE.RKA	3264	6300	0077	FIXED
IMAGE.RKB	3204	6300	0077	FIXED
IMAGE.RKC	3264	6300	0077	FIXED
IMAGE.RKD	3264	6300	0077	FIXED
LOGIN.SAV	9	11	5152	FIXED
OS8.OS8	640	1200	0052	FIXED
OS8.RTS	1722	3300	1152	VARIABLE
SWAPTRA.CKS			7777	FIXED

The file FREEBLO.CKS contains a list of all free areas on the disk. A free area is an area which has not been assigned to an ETOS file. The program FREE (see 8.6) allows you to inspect this file. When you obtain a list of ETOS files, the total number of free blocks is printed at the end of the list. Each account added to the system eliminates a minimum of two blocks from the free block pool. One block is used for the user file directory, which contains a list of ETOS files in the account. The other block is a map block which is used to locate the user file directory segments. Each variable length file created in an account eliminates one block from the free block pool in addition to the number of blocks allocated to the file. This block is used as a map block to locate the ETOS file segments. A fixed length file does not require a map block because it is contiguous. The free blocks file must not be deleted.

The file IMAGE.RK5 is a dummy file, which takes up no space on the disk. This dummy file allows you to access the entire system disk. Under normal operating conditions, you could only access the entire disk by accessing each ETOS file on the disk. This

file is used by the DKCOPY program (see 6.6), which allows you to copy the system disk while ETOS is running. The size of IMAGE.RK5 on the System Industries disk is more than twice the size of IMAGE.RK5 on the RKØ5, due to the larger disk. The IMAGE.RK5 file must not be deleted.

On an RKØ5 disk, there are two additional dummy files called IMAGE.RKA and IMAGE.RKB. On a System Industries disk, there are four additional dummy files called IMAGE.RKA, IMAGE.RKB, IMAGE.RKC, IMAGE.RKD. The purpose of these files is similar to IMAGE.RK5, since they allow you to access the entire system disk. However, these IMAGE files allow you to access the system disk in portions of 3248 or 3264 blocks, instead of one file of 6496 or 13056 blocks. The reason for this segmentation is that FUTIL will not allow you to access an area larger than 4096. If you wanted to verify the system disk for bad blocks (see 8.4.3), you can verify it, one IMAGE file at a time. Table 5-14 contains the system disk blocks represented by the IMAGE files.

Table 5-14
Image File Layouts

IMAGE File	RKØ5 Blocks (octal)	System Industries Blocks (octal)
IMAGE.RKA	Ø-6257	Ø-6277
IMAGE.RKB	6260-14537	6300-14577
IMAGE.RKC		14600-23077
IMAGE.RKD		23100-31377

The IMAGE files must not be deleted.

The file LOGIN.SAV contains a program which allows you to LOGIN to or LOGOUT from the ETOS system. The source copy of LOGIN is provided on the distribution pack in account [Ø,7]. You can modify the source, compile it, save it as a core image file and transfer the file to account [Ø,2]. This process allows you to tailor the LOGIN process to your site. See Section 5.12.3 for

additional details.

The file OS8.OS8 contains the stand-alone or real version of the OS/8 operating system and programs. This file points to the same area as the stand-alone OS/8 RKAØ handler. If you wish to transfer a file to the stand-alone protion of the ETOS pack, the file OS8.OS8 is utilized. See 5.13.6 for further details on this transfer. OS8.OS8 must not be deleted.

The file OS8.RTS contains the timesharing or virtual version of the OS/8 operating system and programs. This area is accessible to all users upon LOGIN as the device SYS:. Core image files are run from this area by entering R program. Since this area is shared by all users, it is write protected. A privileged user can write into this area by write enabling it (see 5.13.7). Virtual OS/8 must not be deleted but it may be extended or reduced.

The file SWAPTRA.CKS contains the area which users are swapped into or out of, when there is not enough room to keep all users in memory. The swap tracks are created by the Sysgen option of ETOS (see 4.2). The size of the file varies according to the number of jobs and the size of each job. SWAPTRA.CKS must not be deleted.

There are several files which are created in account [Ø,2] by the program OSETOS (see 5.12). These files reside on the virtual OS/8 (SYS) portion of the pack. The OSETOS program transfers these files from virtual OS/8 to ETOS files in account [Ø,2]. A list of these files is contained in Table 5-15.

Table 5-15
Additional Library Contents

ETOS filename	Length (Decimal)	Length (Octal)	Protection Code	Fixed or Variable
DKCOPY.SAV	3	3	ØØ5Ø	FIXED
INSTALL.TXT	1	1	ØØ5Ø	FIXED
LOGMESG.TXT	1	1	ØØ5Ø	FIXED
SYSTAT.SAV	12	1Ø	4Ø5Ø	FIXED

DKCOPY.SAV is a program which copies ETOS disks while the system is running. See 5.12.2 for additional information about creating this file.

INSTALL.TXT is a file which contains the installation name printed by SYSTAT. See 5.12.4 for additional information about creating this file.

LOGMESG.TXT is a file which contains the system-wide LOGIN message. See 5.12.5 for additional information about creating this file.

SYSTAT.SAV is a program which provides you with a system status, consisting of all users who are LOGged INto ETOS and their status. See 5.12.1 for additional information about creating this file.

5.15.3 Operator's Account (Account [ØØ,Ø3])

Account [ØØ,Ø3] is the operator's account. The password for account [ØØ,Ø3] is "OPERATOR". After Sysgening a disk, it is the only account you can LOGIN to. All operator functions (e.g., initializing the time and date, creating accounts) are performed in this account. The operator's account may only be entered by a user on the console terminal. If a user attempts to LOGIN to account [Ø,3] on a terminal other than the console, the message "?PROTECTION VIOLATION" is printed and access is denied. Table

5-16 contains a list of ETOS files in account [Ø,3] on a newly Sysgened disk.

Table 5-16
Operator's Account Contents

ETOS Filename	Length (Decimal)	Length (Octal)	Protection Code	Fixed or Variable
JOBØ3.SBK	24	3Ø	ØØ25	FIXED
JOBØ4.SBK	24	3Ø	ØØ25	FIXED
JOBØ5.SBK	24	3Ø	ØØ25	FIXED
JOBØ6.SBK	24	3Ø	ØØ25	FIXED
JOBØ7.SBK	24	3Ø	ØØ25	FIXED
JOB1Ø.SBK	24	3Ø	ØØ25	FIXED
JOB11.SBK	24	3Ø	ØØ25	FIXED
JOB12.SBK	24	3Ø	ØØ25	FIXED
JOB13.SBK	24	3Ø	ØØ25	FIXED
JOB14.SBK	24	3Ø	ØØ25	FIXED
JOB15.SBK	24	3Ø	ØØ25	FIXED
JOB16.SBK	24	3Ø	ØØ25	FIXED
JOB17.SBK	24	3Ø	ØØ25	FIXED
JOB2Ø.SBK	24	3Ø	ØØ25	FIXED
JOB21.SBK	24	3Ø	ØØ25	FIXED
JOB22.SBK	24	3Ø	ØØ25	FIXED

When ETOS is started up, two system jobs are automatically initiated. The ETOS command processor, SCALE (see Chapter 3, System User's Guide) is initiated as job number Ø1. The ETOS disk monitor, DMON is initiated as job number Ø2. The first user who LOGs INto ETOS is assigned job number Ø3. The second user is assigned job number Ø4. Job numbers are assigned sequentially until job number 22. If a user LOGs OUT, their job number is assigned to the next user who LOGs IN. Each job requires a scratch block file. This scratch block file contains the OS/8 scratch blocks for the job and the OS/8 batch monitor scratch blocks. Since it is possible to run jobs which are DETACHed from a terminal, you may run more jobs than the number of terminals on the system. A newly SYSgened pack contains scratch blocks for sixteen jobs. If you wish to run less than sixteen jobs, you can gain additional disk space by deleting scratch block files you do not need. The first file which is deleted is JOB22.SBK. The

second file which is deleted is JOB21.SBK. Continue deleting files, decrementing the job number by one, until you have the desired number of scratch blocks remaining. See 5.6 for additional information about deleting ETOS files.

There are several files which are created by various ETOS programs in account [0,3]. Table 5-17 contains a list of these files.

Table 5-17
Additonal Operator Files

ETOS filename	Length (Decimal)	Length (Octal)	Protection Code	Fixed or Variable
ACCTTMX.YYY	1	1	0025	FIXED
XXXXSPL.QUE	VARIES	VARIES	0050	VARIABLE
TEMPXX.LPT	VARIES	VARIES	0074	VARIABLE

Files in the form ACCTTMX.YYY are created or updated by LOGIN.SAV when any user LOGs OUT. Cumulative usage statistics for the accounts are maintained in these files. In the general file name, X is the first digit of the account number and YYY represents the last digit of the account number. Therefore, the statistics file for account [0,3] is called ACCTTM0.003. If an account is deleted, the statistics file for the account may be deleted to acquire additional free blocks. The cumulative statistics in these ACCTIM files are printed out by the USAGE program (see 6.4.14).

Files in the forms XXXXSPL.QUE are created by the initiation of a line printer or terminal spooler (see 6.4.13). XXXX represents the logical device name of the spooler, which is specified when the spooler is initiated. The logical name is normally "LP00". Therefore, the file name created in account [0,3] is normally called LP00SPL.QUE. This spooler file is a file queue, which contains the list of files waiting to print.

Files in the form TEMPXX.LPT are created by the OS/8 spooling handler, QLP (see 4.9, System User's Guide). These files are created on the system disk in the account of the user requesting spooling. "XX" starts at "ØØ" and is incremented by 1. The first file spooled by a user in an account is of the form TEMPØØ.LPT. The second file spooled is of the form TEMPØ1.LPT. Once these files are listed, the temporary files are deleted and the names can be used again.

5.16 DISTRIBUTION ACCOUNTS DOCUMENTATION

In addition to the accounts created by the ETOS Sysgen option, the distribution pack contains additional accounts. These accounts contain reference material which is not necessary for daily operation of ETOS. The reference accounts are deleted from the pack by the Sysgen. If you want to access the information, you can make a copy of the distribution pack and run it as a system pack or mount it in a non-system drive. Table 5-18 contains a list of the additional distribution accounts.

Table 5-18
Distribution Accounts

Account Number	Password	Protection Code
[ØØ,04]	DOCUMEN	6Ø77
[ØØ,05]	DIFFER	6Ø77
[ØØ,06]	DEVICES	6Ø77
[ØØ,Ø7]	ETOSCSP	6Ø77
[ØØ,1Ø]	ASSEMBL	6Ø77
[ØØ,11]	STANDAL	6Ø77

Each account contains an OS/8 storage area called OS8DISK.DSK. The storage area is a variable length file with a protection code of ØØ5Ø. The length varies according to the amount of information stored. A list of the contents of each accounts OS8DISK.DSK area is contained in the following subsections.

5.16.1 Documentation (Account [00,04])

The OS/8 account storage area for account [00,04] contains documentation files for various software availble under ETOS. The password for account [00,04] is "DOCUMEN". To list one of those files on the line printer, LOGIN to account [0,4] on a copy of the distribution pack and enter LIST file.ex. To list a file on the terminal, enter TYPE file.ex. Table 5-19 contains a list of files contained in the documentation storage area and an explanation of these files.

Table 5-19
Documentation Contents

Filename	Description
COBMAN.WR	Operating instructions for ETOS COBOL. It is an addendum to the COBOL Manual.
COBOL.WR	List of documentation changes to the COBOL Manual.
FOCAL.WR	Manual describing the operation of ETOS FOCAL.
LPTPAT.WR	Documentation which allows you to send COBOL compiler messages to the line printer, instead of the terminal.
RUNOFF.WR	Manual describing the operation of the text processor called RUNOFF.SV.

5.16.2 Differences (Account [00,05])

Account [0,5] contains PAL8 assembler source files which are used to convert the OS/8 programs to run under ETOS. The password for account [00,05] is "DIFFER". ETOS versions of OS/8 programs are created by loading the OS/8 core image file into memory, overlaying it with the binary version of the assembler file corresponding to the program and saving the resultant core image. This process implies that the patches which are released in the DEC software news can be applied without modification to the ETOS

versions of the program, if the patches do not conflict with the ETOS patches. A sample creation of the ETOS version of PAL8 is contained in Figure 5-83.

Figure 5-83
Sample Creation of ETOS Cusp

```
.COMPILE PAL8.DI

ERRORS DETECTED: Ø
LINKS GENERATED: Ø
.R ABSLDR
*SYS:PAL8.SV/I
*PAL8.BN$                    ($ - escape or alt mode)
.SAVE SYS:PAL8
.
```

The ETOS patch files are of the form filename.DI, where filename represents the OS/8 program modified by the overlay. Table 5-2Ø contains a list of these patch files.

Table 5-2Ø
ETOS Differences

ABSLDR.DI	BATCH.DI	BITMAP.DI	BUILD.DI
CBASIC.DI	CBCOMP.DI	CBLOAD.DI	CBRTS.DI
CCL.DI	COBOL9.DI	CREF.DI	DIRECT.DI
EDIT.DI	EPIC.DI	FORLIB.DI	FORT.DI
FOTP.DI	FRTS.DI	FUTIL.DI	F4.DI
HELP.DI	LIBRA.DI	LIBSET.DI	LIB8.DI
LINK.DI	LOAD.DI	LOADER.DI	MACREL.DI
OBASIC.DI	OBCOMP.DI	OBLOAD.DI	OBRTS.DI
PAL8.DI	PASS2.DI	PASS2Ø.DI	PASS3.DI
PIP.DI	RALF.DI	RESORC.DI	SABR.DI
SET.DI	SRCCOM.DI	TECO.DI	

5.16.3 Device Handlers (Account [ØØ,Ø6])

Account [Ø,6] contains the source to all of the OS/8 device handlers which have been modified to run under ETOS. The password for account [ØØ,Ø6] is "DEVICE". These device handlers are LOADed into the ETOS version of BUILD. Not all of these handlers are activated in the distribution system. OS/8 has a

limit of fifteen active handlers. To list active handlers or activate different handlers, see 6.4.1. The ETOS version of BUILD was created by unLOADing all handlers and then LOADing in the ETOS versions of the handlers as illustrated in Figure 5-84.

Figure 5-84
Sample Load of ETOS Device Handler

```
.COMPILE ETCR.PA

ERRORS DETECTED:Ø
LINKS GENERATED:Ø
.RUN SYS:BUILD
$LOAD ETCR
$
```

Table 5-21 contains a list of all ETOS OS/8 device handlers, their group name and the handlers in the group. Further information about these handlers is provided in 6.4.1.

Table 5-21
ETOS OS/8 Device Handlers

Filename	Group Name	Handlers in Group
BAT.PA	BAT	BAT
DUMP.PA	DUMP	DUMP
ETCDR.PA	ETCR	CDR
ETKL8E.PA	ETKL	TTY
ETLPT.PA	ETLP	LPT
ETLSPT.PA	ETLS	PTA PTR
ETLT.PA	ETLT	LT
ETOSRK.PA	ETOS	SYS
ETOSSI.PA	ETOS	SYS DSKØ DSK1 DSK2 DSK3
ETOSSY.PA	ETOS	SYS CHNØ CHN1 CHN2 CHN3 CHN4 CHN5 CHN6 CHN7
ETPT8E.PA	ETPT	HSP HSR
ETRXNS.PA	ETRX	RXAØ RXA1
ETTD8E.PA	ETTD	DTAØ DTA1
ETTY.PA	ETTY	TTY
ETSPLP.PA	SPLP	QLP
ETTCØ1.PA	TCØ1	DTAØ DTA1
ETTC23.PA	TC23	DTA2 DTA3
ETTC45.PA	TC45	DTA4 DTA5
ETTC67.PA	TC67	DTA6 DTA7

5.16.4 ETOS Cusps (Account [00,07])

Account [0,7] contains the PAL8 source to all Commonly Used System Programs (CUSPs) written expressly for ETOS. the password for account [00,07] is "ETOSCSP". The only source which are is included is the ACCNT source. It is not included due to its proprietary nature. The source is included in the ETOS source kit, which may be purchased as an additional option. The assembler sources may be converted to the version of the program which is contained in virtual OS/8 by using the dialogue contained in Figure 5-85.

Figure 5-85
Compilation of ETOS Cusps

```
.^VPRIV 4040
^VWE 0
.COMPILE filename.PA

ERRORS DETECTED:0
LINKS GENERATED:0
.LOAD filename
.SAVE SYS:filename
```

If you are changing the source to LOGIN, you must take the additional step of running OSETOS (see 5.12) to transfer LOGIN.SV to the ETOS file [0,2] LOGIN.SAV.

Table 5-22 contains a list of all sources contained in account [0,7].

Table 5-22
ETOS CUSP Source Files

Source File Name	ETOS CUSP Name	REAL or Virtual OS/8
BACKUP.PA	BACKUP.SV	REAL
BREAK.PA	BREAK.CS	VIRTUAL
CDUMP.PA	CDUMP.SV	REAL
CONFIG.PA	CONFIG.SV	REAL
COSBLD.PA	COSBLD.SV	VIRTUAL
COSBO.PA	COSBO.SV	VIRTUAL
DKCOPY.PA	DKCOPY.SV	VIRTUAL
DSKINT.PA	DSKINT.SV	VIRTUAL
ERRCPY.PA	ERRCPY.SV	VIRTUAL
FREE.PA	FREE.SV	VIRTUAL
INIT.PA	INIT.SV	VIRTUAL
INQUIR.PA	INQUIR.SV	VIRTUAL
LOGIN.PA	LOGIN.SV	VIRTUAL
LOOKUP.PA	LOOKUP.SV	VIRTUAL
OS8BO.PA	OS8BO.CS	VIRTUAL
OSETOS.PA	OSETOS.SV	VIRTUAL
RKCOPY.PA	RKCOPY.SV	REAL
SHUTUP.PA	SHUTUP.SV	VIRTUAL
SICOPY.PA	SICOPY.SV	REAL
SPOOLR.PA	SPOOLR.SV	VIRTUAL
SYSTPT.PA	SYSTAT.SV	VIRTUAL
TDSTOP.PA	TDSTOP.SV	VIRTUAL
TIME.PA	TIME.SV	VIRTUAL
TTYSET.PA	TTYSET.SV	VIRTUAL
UNBRK.PA	UNBRK.CS	VIRTUAL
USAGE.PA	USAGE.SV	VIRTUAL
WHO.PA	WHO.SV	VIRTUAL

5.16.5 Assembly Language (Account [00,10])

Account [0,10] contains information which is helpful in writing assembly language code under ETOS and implementing non-standard versions of ETOS-supported peripherals. The password for account [00,10] is "ASSEMBL". Table 5-23 contains a list of these files.

Table 5-23
Assembly Language Informational Files

File Name	Description
CARD.LS	Listing of ETOS internal card reader handler.
CLOCK.LS	Listing of ETOS internal clock handler.
EDEFS.PA	PAL8 source file containing definitions for ETOS IOTs, discussed in 6.4, System User's Guide and 6.3, System Manager's Guide.
LPSERV.LS	Listing of ETOS internal line printer handler.
MULDIV.MA	Assembly language subroutine, which performs double precision multiplication and division.
PRVFNC.PA	Sample use of real-time programming under ETOS.
RKØ5.LS	Listing of ETOS internal RKØ5 disk handlers.
RTDEMO.PA	PAL8 source file of real-time demonstration program (see 7.2, System User's Guide).
SI3Ø4Ø.LS	Listing of ETOS internal System Industries 3Ø/4Ø disk handler.

All files listed above can be printed on the line printer by LOGging INto account [Ø,1Ø] on a copy of the distribution pack and entering LIST filename.extension. A listing could be obtained on the terminal by entering TYPE filename.extension. All files except EDEFS.PA and RTDEMO.PA are informational files and can only be listed. EDEFS.PA can be compiled with an ETOS program which does not contain definitions by entering COMPILE program<EDEFS, program. RTDEMO.PA can be compiled, loaded and saved in a core image file with the dialogue contained in Figure 5-86.

Figure 5-86
Sample Creation of Assembler Core Image File

```
.COMPILE RTDEMO.PA

ERRORS DETECTED:Ø
LINKS GENERATED:Ø

.LOAD RTDEMO
.SAVE DSK:RTDEMO
.
```

5.16.6 Stand-alone Software

Account [Ø,11] contains the stand-alone versions of all programs which have been modifed for use under ETOS. The password for account [ØØ,11] is STANDAL. These programs can be utilized to build a stand-alone OS/8 pack. For some products, not all modules have to be modified. For example, in COBOL, only COBOL9.RL is modified to run under ETOS. Therefore, only COBOL9.RL is contained in account [Ø,11]. To make up a stand-alone version of COBOL, you must copy all COBOL modules from virtual OS/8 SYS and overlay COBOL9.RL with the file contained in this account. Creation of the stand-alone version of each product is discussed in section 5.1Ø.3.

All of the modules contained in account [Ø,7] are documented in the OS/8 Handbook. Table 5-24 contains a list of all stand-alone modules contained in account [Ø,11].

Table 5-24
Stand-alone Files

ABSLDR.SV	ASR33.BN	ATAN.RL	BASIC.SV	BAT.BN
BATCH.SV	BITMAP.SV	BOOT.SV	BUILD.SV	CAMP.SV
CBASIC.SV	CBCOMP.SV	CBLOAD.SV	CBRTS.SV	CCL.PA
CCL.SV	COBOL9.RL	CREF.SV	CR8E.BN	CSA.BN
CSB.BN	CSC.BN	CSD.BN	DF32NS.BN	DF32SY.BN
DIRECT.SV	DTCOPY.SV	DTFRMT.SV	DUMP.BN	EABRTS.BN
EDIT.SV	EPIC.SV	FOCAL.SV	FORLIB.RL	FORT.SV
FOTP.SV	FRTS.SV	FUTIL.SV	F4.SV	HELP.HL
HELP.SV	KL8E.BN	KL8E.PA	KREF.SV	KRETRV.SV
KSETUP.SV	KSHUFL.SV	KSORT.SV	LIB8.RL	LINCNS.BN
LINCSY.BN	LOAD.SV	LOADER.SV	LPSV.BN	LQP.BN
LSPT.BN	L645.BN	MACREL.SV	MCPIP.SV.	MSBAT.SV
OBASIC.AF	OBASIC.FF	OBASIC.SF	OBASIC.SV	OBASIC.UF
OBCOMP.SV	OBLOAD.SV	OBRTS.SV	PAL8.SV	PASS2.SV
PASS28.SV	PASS3.SV	PIP.SV	PIP1Ø.SV	PT8E.BN
RALF.SV	RESORC.SV	RF08NS.BN	RF08SY.BN	RKLFMT.SV
RK08NS.BN	RK08SY.BN	RK8ENS.BN	RK8ESY.BN	ROMMSY.BN
RXCOPY.SV	RX01NS.BN	RX01SY.BN	SABR.SV	SET.SV
SRCCOM.SV	TC08NS.BN	TC08SY.BN	TDCOPY.SV	TDFRMT.SV
TDINIT.SV	TD8EA.BN	TD8EB.BN	TD8EC.BN	TD8ED.BN
TD8ESY.BN	TECO.BN	TECO.SV	TM8E.BN	VR12.BN
VT50.BN				

Not all users will have all of the programs listed in Table 5-24. You have only the programs for which you are licensed. Section 1.6 contains a list of all licensed products and the programs distributed with the licenses.