J 6

```
MMM         MMM  TTTTTTTTTTTTTTT     AAAAAAAAA        AAAAAAAAA        CCCCCCCCCCCC  PPPPPPPPPPPP
MMM         MMM  TTTTTTTTTTTTTTT     AAAAAAAAA        AAAAAAAAA        CCCCCCCCCCCC  PPPPPPPPPPPP
MMM         MMM  TTTTTTTTTTTTTTT     AAAAAAAAA        AAAAAAAAA        CCCCCCCCCCCC  PPPPPPPPPPPP
MMMMMM   MMMMMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP         PPP
MMMMMM   MMMMMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP         PPP
MMMMMM   MMMMMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP         PPP
MMM   MMM   MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP         PPP
MMM   MMM   MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP         PPP
MMM   MMM   MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP         PPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPPPPPPPPPPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPPPPPPPPPPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPPPPPPPPPPP
MMM         MMM        TTT        AAAAAAAAAAAAAAA  AAAAAAAAAAAAAAA  CCC              PPP
MMM         MMM        TTT        AAAAAAAAAAAAAAA  AAAAAAAAAAAAAAA  CCC              PPP
MMM         MMM        TTT        AAAAAAAAAAAAAAA  AAAAAAAAAAAAAAA  CCC              PPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA     CCCCCCCCCCCC  PPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA     CCCCCCCCCCCC  PPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA     CCCCCCCCCCCC  PPP
```

```
                                                                      M 16
**FILE**ID**HEADER





HH      HH  EEEEEEEEEE    AAAAAA    DDDDDDDD    EEEEEEEEEE  RRRRRRR
HH      HH  EEEEEEEEEE    AAAAAA    DDDDDDDD    EEEEEEEEEE  RRRRRRR
HH      HH  EE          AA      AA  DD      DD  EE          RR      RR
HH      HH  EE          AA      AA  DD      DD  EE          RR      RR
HH      HH  EE          AA      AA  DD      DD  EE          RR      RR
HH      HH  EE          AA      AA  DD      DD  EE          RR      RR
HHHHHHHHHH  EEEEEEEE    AA      AA  DD      DD  EEEEEEEE    RRRRRRR
HHHHHHHHHH  EEEEEEEE    AA      AA  DD      DD  EEEEEEEE    RRRRRRR
HH      HH  EE          AAAAAAAAAA  DD      DD  EE          RR  RR
HH      HH  EE          AAAAAAAAAA  DD      DD  EE          RR  RR
HH      HH  EE          AA      AA  DD      DD  EE          RR    RR        ....
HH      HH  EE          AA      AA  DD      DD  EE          RR    RR        ....
HH      HH  EEEEEEEEEE  AA      AA  DDDDDDDD    EEEEEEEEEE  RR      RR      ....
HH      HH  EEEEEEEEEE  AA      AA  DDDDDDDD    EEEEEEEEEE  RR      RR      ....

LL            IIIIII      SSSSSSSS
LL            IIIIII      SSSSSSSS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II        SSSSSS
LL              II        SSSSSS
LL              II              SS
LL              II              SS
LL              II              SS
LL              II              SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

```
0
0 MODULE HEADER (LANGUAGE (BLISS32) ,
0                IDENT = 'V04-000'
0                ) =
1 BEGIN
1 !
1 !****************************************************************************
1 !*                                                                          *
1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
1 !*  ALL RIGHTS RESERVED.                                                    *
1 !*                                                                          *
1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
1 !*  TRANSFERRED.                                                            *
1 !*                                                                          *
1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
1 !*  CORPORATION.                                                            *
1 !*                                                                          *
1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
1 !*                                                                          *
1 !*                                                                          *
1 !****************************************************************************
1
1 !++
1 !
1 ! FACILITY:  MTAACP
1 !
1 ! ABSTRACT:
1 !       This module contains routines which position to headers or trailers
1 !       and read them.
1 !
1 ! ENVIRONMENT:
1 !
1 !       Starlet operating system, including privileged system services
1 !       and internal exec routines.
1 !
1 !--
1
1 !
1 !
1 ! AUTHOR:  D. H. GILLESPIE,      CREATION DATE:  25-MAY-77 15:00
1 !
1 ! MODIFIED BY:
1 !
1 !       V03-006 MMD0323         Meg Dumont,     13-Aug-1984  15:17
1 !               Fix to fix MMD0285, the way it was implemented the call
1 !               wasn't getting made.
1 !
1 !       V03-005 MMD0300         Meg Dumont,     20-Jun-1984  11:23
1 !               Fix to default Buffer offset length to zeros, when no HDR2
1 !               is present for the file.
```

```
1 !
1 !     V03-004 MMD0285         Meg Dumont,     6-Apr-1984  17:18
1 !             Fix to READ_HDR to incllude calling the clear
1 !             seriuos exception routine after the headers are
1 !             read. This is so that we do not leave the
1 !             TMSCP drives left in serious exception state
1 !             if we read into the TM while reading the headers.
1 !
1 !     V03-003 MMD0280         Meg Dumont,     23-Mar-1984  10:27
1 !             Fix long file name support such that for ANSI version
1 !             3 volumes it converts the exentsion length to
1 !             ASCII characters before writing it to the label.
1 !
1 !     V03-002 ROW0258         Ralph O. Weber          21-NOV-1983
1 !                     The Paul Painter Memorial Enhancement
1 !             Named for one of the unfortunate customers who suffered much
1 !             to determine the great UCB$L_MT_RECORD secret while trying to
1 !             create a user-written magtape driver, this change eliminates
1 !             use of the device dependent field, UCB$L_MT_RECORD in favor of
1 !             the device independent field, UCB$L_RECORD.
1 !
1 !     V03-001 MMD0162         Meg Dumont,     26-Apr-1983  9:36
1 !             Change reference to 80 to the s,mbol ANSI_LBLSZ. Change READ_HDR
1 !             to read in the HDR4 label or if not found to default the values.
1 !
1 !     V02-010 REFORMAT        Maria del C. Nasr       30-Jun-1980
1 !
1 !     V02-009 MCN0016         Maria del C. Nasr,      18-Jun-1980  11:55
1 !             Initialize default HDR2 with blanks, instead of zeroes, to
1 !             avoid setting the old RMS attributes field.
1 !
1 !     A0008   MCN0013         Maria del C. Nasr       11-Mar-1980  11:25
1 !             Check for HDR3 instead of HDR2 to determine if current file
1 !             should be includeo in search or not.
1 !
1 !     A0007   MCN0011         Maria del C. Nasr       04-Feb-1980  9:05
1 !             Add input parameter to UPDVCB_LEOV routine to either clear
1 !             or set flag, and make routine global.
1 !
1 !     A0006   MCN0003         Maria del C. Nasr       28-Sep-79   10:39
1 !             Add HDR3 processing
1 !
1 !     A0005   SPR24948        Maria del C. Nasr       11-Sep-79       17:30
1 !             Forced spacing to eof when current position bit set to
1 !             fix bug.
1 !
1 !
1 !**
1
1 LIBRARY 'SYS$LIBRARY:LIB.L32';
1
1 REQUIRE 'SRC$:MTADEF.B32';
1
1 FORWARD ROUTINE
1     READ_HDR            : COMMON_CALL,          ! read HDR1, HDR2, and HDR3 and HDR4 if exist
1     SPACE_EOF           : COMMON_CALL NOVALUE,  ! space to end of file
1     SET_NUMBER_OF_LABELS : COMMON_CALL NOVALUE, ! set the number of labels read
```

```
1      UPDVCB_LEOV        : COMMON_CALL NOVALUE,  ! update VCB logical end of file
1      MAKE_COR FILE      : COMMON_CALL NOVALUE,  ! update VCB
1      WRAP_AROUND        : L$WRAP_AROUND;        ! continue search at beginning of volume set  !
1
1 EXTERNAL
1      CURRENT_UCB        : REF BBLOCK,
1      IO_PACKET          : REF BBLOCK,           ! address of IO request packet
1      HDR1               : REF BBLOCK,           ! address HDR1 label
1      HDR2               : REF BBLOCK,           ! address of HDR2 label
1      HDR3               : REF BBLOCK,           ! address of HDR3 label
1      HDR4               : REF BBLOCK;           ! address of HDR4 label
1
```

```
1 GLOBAL ROUTINE GET_START_HDR : L$GET_START_HDR =
1
1 !++
1 !
1 ! FUNCTIONAL DESCRIPTION:
1 !       This routine positions to the header label set of the start file
1 !       in current search and reads HDR1, HDR2, HDR3 and HDR4 labels unless
1 !       they have already been read.
1 !
1 ! CALLING SEQUENCE:
1 !       GET_START_HDR()
1 !
1 ! INPUT PARAMETERS:
1 !       none
1 !
1 ! IMPLICIT INPUTS:
1 !       CURRENT_VCB, CURRENT_UCB
1 !
1 ! OUTPUT PARAMETERS:
1 !       none
1 !
1 ! IMPLICIT OUTPUTS:
1 !       HDR1 read in, HDR2 read in or defaulted, HDR3 read in or defaulted
1 !       HDR4 read in or defaulted
1 !
1 ! ROUTINE VALUE:
1 !       0 unsuccessful, logical end of volume set
1 !       1 successful
1 !
1 ! SIDE EFFECTS:
1 !       none
1 !
1 !--
1
2     BEGIN
2
2     EXTERNAL REGISTER
2         COMMON_REG;
2
2     EXTERNAL ROUTINE
2         MOUNT_VOL       : COMMON_CALL;          ! mount volume
2
2     EXTERNAL
2         CURRENT_UCB     : REF BBLOCK,           ! address of current ucb
2         LOCAL_FIB       : BBLOCK;                       ! copy of user's fib
2
2     LOCAL
2         RELATIVE_BLOCK,                 ! relative block number to last tm
2         TM;                                     ! number of tm's
2
2     ! mount volume if the current relative volume number is zero
2     !
2
2     IF .CURRENT_VCB[VCB$B_CUR_RVN] EQL 0
2     THEN
2         MOUNT_VOL(1, $FIELDMASK(MOU$V_REWIND) + $FIELDMASK(MOU$V_LBLCHECK));
2
```

```
2      ! if at logical end of volume set, return immediately
2      !
2
2      IF .CURRENT_VCB[VCB$V_LOGICEOVS]
2      THEN
2          RETURN 0;
2
2      ! If the number of tape marks into the file is not 0, then the previous file
2      ! was closed prematurely and should not be included in search except in the
2      ! case where there is no HDR3 and the tape is left positioned beyond the
2      ! tm. If the section is not the first, then space to next file
2      !
2
3      IF (.CURRENT_VCB[VCB$B_TM] NEQU 0
3          AND
4          NOT (.CURRENT_VCB[VCB$B_TM] EQLU 1 AND .HDR3[HD3$L_HD3LID] NEQU 'HDR3'
4          AND
3          (.CURRENT_UCB[UCB$L_RECORD] - .CURRENT_VCB[VCB$L_ST_RECORD]) EQLU 0))
2          OR
2          .CURRENT_VCB[VCB$W_CUR_SEQ] GTR 1
2      THEN
2          SPACE_EOF()                              ! position to beginning of next file
2      ELSE
2
2          ! If function is create, and current position bit is set, then force
2          ! spacing to end of file, unless positioned in dummy file header set...
2          !
2
3          IF ((.IO_PACKET[IRP$V_FCODE] EQL IO$_CREATE) AND .LOCAL_FIB[FIB$V_CURPOS])
2              AND
3              (.CURRENT_VCB[VCB$B_TM] NEQU 0) AND (.CURRENT_VCB[VCB$W_CUR_NUM] NEQU 0)
2          THEN
2              SPACE_EOF();
2
2      ! When new volume is mounted, VOL1 has been read but not the header labels.
2      ! Therefore, the actual block count equals 1.  If relative block count = 0,
2      ! then the headers have not been read for this file.
2      !
2      RELATIVE_BLOCK = .CURRENT_UCB[UCB$L_RECORD] - .CURRENT_VCB[VCB$L_ST_RECORD];
2
3      IF (.RELATIVE_BLOCK EQL 0 OR .CURRENT_UCB[UCB$L_RECORD] EQLU 1)
2          AND
2          .CURRENT_VCB[VCB$B_TM] EQLU 0
2      THEN
2          RETURN READ_HDR();
2
2      RETURN 1;
2
1      END;                                         ! end of routine

                                                      .TITLE  HEADER
                                                      .IDENT  \V04-000\

                                                      .EXTRN  CURRENT_UCB, IO_PACKET
                                                      .EXTRN  HDR1, HDR2, HDR3
                                                      .EXTRN  HDR4, MOUNT_VOL
```

```
                                                                           .EXTRN   LOCAL_FIB

                                                                           .PSECT   $CODE$,NOWRT,2

                                                      5A  DD 00000 GET_START_HDR::
                                                                           PUSHL    R10                                          ; 0510
                                                 2F   AB  95 00002         TSTB     47(CURRENT_VCB)                              ; 0563
                                                      09  12 00005         BNEQ     1$                                           ;
                                                      03  DD 00007         PUSHL    #3                                           ; 0565
                                                      01  DD 00009         PUSHL    #1                                           ;
                                0000G   CF            02  FB 0000B         CALLS    #2, MOUNT_VOL                                ;
                         73        0B   AB            01  E0 00010 1$:     BBS      #1, 11(CURRENT_VCB), 7$                      ; 0570
                                                      5A  D4 00015         CLRL     R10                                          , 0580
                                                 2E   AB  95 00017         TSTB     46(CURRENT_VCB)                              ;
                                                      20  13 0001A         BEQL     2$                                           ;
                                                      5A  D6 0001C         INCL     R10                                          ;
                                        01       2E   AB  91 0001E         CMPB     46(CURRENT_VCB), #1                          ; 0582
                                                      39  12 00022         BNEQ     3$                                           ;
                             33524448   8F    0000G   DF  D1 00024         CMPL     @HDR3, #861029448                            ;
                                                      2E  13 0002D         BEQL     3$                                           ;
                                        50    0000G   CF  D0 0002F         MOVL     CURRENT_UCB, R0                              ; 0584
                                   30   AB     00B0   C0  D1 00034         CMPL     176(R0), 48(CURRENT_VCB)                     ;
                                                      21  12 0003A         BNEQ     3$                                           ;
                                        01       26   AB  B1 0003C 2$:     CMPW     38(CURRENT_VCB), #1                          ; 0586
                                                      1B  1A 00040         BGTRU    3$                                           ;
                                        50    0000G   CF  D0 00042         MOVL     IO_PACKET, R0                                ; 0595
           33       20   A0             06            00  ED 00047         CMPZV    #0, #6, 32(R0), #51                          ;
                                                      13  12 0004D         BNEQ     4$                                           ;
                         0D     0000G   CF            04  E1 0004F         BBC      #4, LOCAL_FIB, 4$                            ;
                                        0A            5A  E9 00055         BLBC     R10, 4$                                      ; 0597
                                                 24   AB  B5 00058         TSTW     36(CURRENT_VCB)                              ;
                                                      05  13 0005B         BEQL     4$                                           ;
                                0000V   CF            00  FB 0005D 3$:     CALLS    #0, SPACE_EOF                                ; 0599
                                        50    0000G   CF  D0 00062 4$:     MOVL     CURRENT_UCB, R0                              ; 0605
                         51      00B0   C0       30   AB  C3 00067         SUBL3    48(CURRENT_VCB), 176(R0), RELATIVE_BLOCK     ;
                                                      07  13 0006E         BEQL     5$                                           ; 0607
                                        01     00B0   C0  D1 00070         CMPL     176(R0), #1                                  ;
                                                      0C  12 00075         BNEQ     6$                                           ;
                                                 2E   AB  95 00077 5$:     TSTB     46(CURRENT_VCB)                              ; 0609
                                                      07  12 0007A         BNEQ     6$                                           ;
                                0000V   CF            00  FB 0007C         CALLS    #0, READ_HDR                                 ; 0611
                                                      07  11 00081         BRB      8$                                           ;
                                        50            01  D0 00083 6$:     MOVL     #1, R0                                       ; 0613
                                                      02  11 00086         BRB      8$                                           ;
                                                      50  D4 00088 7$:     CLRL     R0                                           ; 0615
                                        5A            8E  D0 0008A 8$:     MOVL     (SP)+, R10                                   ;
                                                          05 0008D         RSB                                                   ;

; Routine Size:  142 bytes,    Routine Base:  $CODE$ + 0000

;  234          0616  1
```

```
1 GLOBAL ROUTINE READ_HDR : COMMON_CALL =
1
1 !++
1 !
1 ! FUNCTIONAL DESCRIPTION:
1 !       Read HDR1, and HDR2 if it exists - otherwise, it is defaulted.
1 !       HDR3 is read only if HDR2 is found, and if starlet file. HDR4
1 !       is read if the HDR3 is read.
1 !
1 ! CALLING SEQUENCE:
1 !       READ_HDR()
1 !
1 ! INPUT PARAMETERS:
1 !       none
1 !
1 ! IMPLICIT INPUTS:
' !       CURRENT_VCB - address of VCB
1 !
1 ! OUTPUT PARAMETERS:
1 !       none
1 !
1 ! IMPLICIT OUTPUTS:
1 !       HDR1, HDR2, HDR3 , and HDR4 read in
1 !       If starlet file, VCB notes this fact
1 !       Also the number of labels that the mtaacp found is set in the VCB
1 !       If logical end of tape (ie:  tm encountered on read of HDR1) then this fact is noted in VCB
1 !
1 ! ROUTINE VALUE:
1 !       0 - tm encountered when reading HDR1, logical end of volume set
1 !       1 - successful
1 !
1 ! SIDE EFFECTS:
1 !       First user label may be located in scratch label area
1 !
1 ! USER ERRORS:
1 !       SS$_TAPEPOSLOST - HDR1 not encountered on read
1 !
1 !--
1
2     BEGIN
2
2     LOCAL
2         MVL     : REF BBLOCK,
2         NUMBER_OF_LABELS,
2         SCRATCH : REF BBLOCK,
2         DESCR   : VECTOR [2,LONG];
2
2     EXTERNAL REGISTER
2         COMMON_REG;
2
2     EXTERNAL ROUTINE
2         CHCK_IO_CLR_EXCP : COMMON_CALL NOVALUE,
2         1SSUE_IO         : L$ISSUE_IO,            ! Issue an IO to tape drive
2         READ_BLOCK       : COMMON_CALL;           ! read one magtape block
2
2     BIND
2         CVT5 = DESCRIPTOR('!5ZW'),
2           DEFAULT = UPLIT ('00512');
2
2       ! Initialize the number of labels read. This number will eventually
2       ! be stored in the VCB and will be used on volume switch and file close
2       ! to determine the number of labels to write to the tape
2
2       NUMBER_OF_LABELS = 0;
2       IF NOT READ_BLOCK(.HDR1, ANSI_LBLSZ)
2       THEN
3           BEGIN
3           KERNEL_CALL(UPDVCB_LEOV, 1);
3           RETURN 0;
3
3           END;
2
2       WHILE 1
2       DO
3           BEGIN
3
3           IF .HDR1[HD1$L_HD1LID] EQLU 'HDR1'
3           THEN
3               EXITLOOP;
3
3           IF NOT READ_BLOCK(.HDR1, ANSI_LBLSZ)
3           THEN
3               ERR_EXIT(SS$_TAPEPOSLOST);
3
2           END;
2
2       NUMBER_OF_LABELS = 1;
2       KERNEL_CALL(MAKE_CUR_FILE, .HDR1);
2
2       ! Default HDR2, HDR3, and HDR4 values
2
2       CH$FILL(' ', ANSI_LBLSZ, .HDR2);
2       CH$FILL(0, ANSI_LBLSZ, .HDR3);              ! clear HDR3 area
2       CH$FILL(' ',ANSI_LBLSZ, .HDR4);             ! clear HDR4 area
2
2       ! Default the HDR4 fields according to the version type.
2
2       MVL = .CURRENT_VCB[VCB$L_MVL];
2       IF .MVL[MVL$B_STDVER] GTR 3
2           THEN
2               HDR4[HD4$B_FILEID_EXT_SIZE] = 0             ! Default size to 0
2           ELSE
2               CH$FILL('0', HD4$S_FILEID_EXT_V3, HDR4[HD4$T_FILEID_EXT_V3]);
2
2       HDR2[HD2$B_RECFORMAT] = 'F';
2       DESCR[0] = HD2$S_BLOCKLEN;
2       DESCR[1] = HDR2[HD2$T_BLOCKLEN];
2
3       IF NOT $FAO(CVT5, 0, DESCR, .CURRENT_UCB[UCB$W_DEVBUFSIZ])
2       THEN
2           CH$MOVE(HD2$S_BLOCKLEN, DEFAULT, HDR2[HD2$T_BLOCKLEN]);
2
2       CH$MOVE(HD2$S_RECLEN, HDR2[HD2$T_BLOCKLEN], HDR2[HD2$T_RECLEN]);
2
```

```
:  350     0731 2    IF .CURRENT_VCB[VCB$W_RECORDSZ] NEQ 0
:  351     0732 2    THEN
:  352     0733 3        BEGIN
:  353     0734 3        DESCR[0] = HD2$S_RECLEN;
:  354     0735 3        DESCR[1] = HDR2[HD2$T_RECLEN];
:  355     0736 3
:  356     0737 4        IF NOT $FAO(CVT5, 0, DESCR, .CURRENT_VCB[VCB$W_RECORDSZ])
:  357     0738 3        THEN
:  358     0739 3            CH$MOVE(HD2$S_RECLEN, HDR2[HD2$T_BLOCKLEN], HDR2[HD2$T_RECLEN]);
:  359     0740 3
:  360     0741 2        END;
:  361     0742 2
:  362     0743 2    ! Set up the default buffer offset length field. In case there
:  363     0744 2    ! is no HDR2 label
:  364     0745 2
:  365     0746 2        HDR2[HD2$T_BUFOFF] = '00';
:  366     0747 2
:  367     0748 2    ! Set up the Scratch area to read the labels into to determine if
:  368     0749 2    ! this is a good label, before copying it into the real label field.
:  369     0750 2
:  370     0751 2    SCRATCH = .HDR1 + SCRATCH_OFFSET;
:  371     0752 2
:  372     0753 2    ! Now try to read HDR2
:  373     0754 2    !
:  374     0755 2
:  375     0756 2    IF READ_BLOCK(.SCRATCH, ANSI_LBLSZ)         ! read into scratch area
:  376     0757 2    THEN
:  377     0758 2
:  378     0759 2        IF .(.SCRATCH) EQLU 'HDR2'
:  379     0760 2        THEN
:  380     0761 3            BEGIN
:  381     0762 3            CH$MOVE(ANSI_LBLSZ, .SCRATCH, .HDR2);       ! HDR2 found
:  382     0763 3            NUMBER_OF_LABELS = 2;
:  383     0764 3
:  384     0765 3            IF .CURRENT_VCB[VCB$V_STARFILE]     ! if starlet file
:  385     0766 3            THEN
:  386     0767 4            BEGIN
:  387     0768 4                IF READ_BLOCK(.SCRATCH , ANSI_LBLSZ)    ! try to read HDR3
:  388     0769 4                THEN
:  389     0770 5                BEGIN
:  390     0771 5                    IF .(.SCRATCH) EQLU 'HDR3'
:  391     0772 5                    THEN
:  392     0773 6                    BEGIN
:  393     0774 6                        CH$MOVE(ANSI_LBLSZ, .SCRATCH, .HDR3);   ! HDR3 found
:  394     0775 6                        NUMBER_OF_LABELS = 3;
:  395     0776 5                    END;
:  396     0777 5                    IF READ_BLOCK(.SCRATCH , ANSI_LBLSZ)    ! try to read HDR4
:  397     0778 5                        THEN
:  398     0779 5                            IF .(.SCRATCH) EQLU 'HDR4'
:  399     0780 5                            THEN
:  400     0781 6                            BEGIN
:  401     0782 6                                CH$MOVE(ANSI_LBLSZ, .SCRATCH, .HDR4);   ! HDR4 found
:  402     0783 6                                NUMBER_OF_LABELS = 4;
:  403     0784 5                            END;
:  404     0785 4                END;
:  405     0786 3            END;
:  406     0787 2            END;
:   407          0788  2
:   408          0789  2      ! Call to clear TMSCP drives of the serious exception (reading the tape
:   409          0790  2      ! mark) before returning to the user
:   410          0791  2
:   411          0792  2      CHCK_IO_CLR_EXCP();
:   412          0793  2      KERNEL_CALL(SET_NUMBER_OF_LABELS, NUMBER_OF_LABELS);
:   413          0794  2      RETURN 1;                                          ! return success
:   414          0795  2
:   415          0796  1      END;                                               ! end of routine



                                57  5A  35  21  0008E P.AAB:  .ASCII   \!5ZW\                                          ;
                                                00092         .BLKB    2
                                      00000004  00094 P.AAA:  .LONG    4                                               :
                                     00000000'  00098         .ADDRESS P.AAB                                           :
                00  00  00  32  31  35  30  30  0009C P.AAC:  .ASCII   \00512\<0><0><0>                                :

                                                      CVT5=                 P.AAA
                                                      DEFAULT=              P.AAC
                                                              .EXTRN   CHCK_IO_CLR_EXCP
                                                              .EXTRN   ISSUE_IO, READ_BLOCK
                                                              .EXTRN   SYS$CMKRNL, SYS$FAO

                                          07FC 00000          .ENTRY   READ_HDR, Save R2,R3,R4,R5,R6,R7,R8,R9,R10      ; 0617
                          5A     0000G  CF  9E 00002          MOVAB    HDR2, R10                                       :
                          59 00000000G  9F  9E 00007          MOVAB    @#SYS$CMKRNL, R9                                :
                          5E            08  C2 0000E          SUBL2    #8, SP                                          :
                                        58  D4 00011          CLRL     NUMBER_OF_LABELS                                ; 0680
                          7E        50  8F  9A 00013          MOVZBL   #80, -(SP)                                      ; 0681
                                 0000G  CF  DD 00017          PUSHL    HDR1                                            :
                   0000G  CF            02  FB 0001B          CALLS    #2, READ_BLOCK                                  :
                          10            50  E8 00020          BLBS     R0, 1$                                          :
                                        01  DD 00023          PUSHL    #1                                              ; 0684
                                        01  DD 00025          PUSHL    #1                                              :
                                        5E  DD 00027          PUSHL    SP                                              :
                                 0000V  CF  9F 00029          PUSHAB   UPDVCB_LEOV                                     :
                          69            04  FB 0002D          CALLS    #4, SYS$CMKRNL                                  :
                                      0165  31 00030          BRW      9$                                              ; 0685
                31524448  8F     0000G  DF  D1 00033 1$:      CMPL     @HDR1, #827475016                               ; 0693
                                        16  13 0003C          BEQL     2$                                              :
                          7E        50  8F  9A 0003E          MOVZBL   #80, -(SP)                                      ; 0697
                                 0000G  CF  DD 00042          PUSHL    HDR1                                            :
                   0000G  CF            02  FB 00046          CALLS    #2, READ_BLOCK                                  :
                          E5            50  E8 0004B          BLBS     R0, 1$                                          :
                                  0224  8F  BF 0004E          CHMU     #548                                            ; 0699
                                        DF  11 00052          BRB      1$                                              ; 0689
                          58            01  D0 00054 2$:      MOVL     #1, NUMBER_OF_LABELS                            ; 0703
                                 0000G  CF  DD 00057          PUSHL    HDR1                                            ; 0704
                                        01  DD 0005B          PUSHL    #1                                              :
                                        5E  DD 0005D          PUSHL    SP                                              :
                                 0000V  CF  9F 0005F          PUSHAB   MAKE_CUR_FILE                                   :
                          69            04  FB 00063          CALLS    #4, SYS$CMKRNL                                  :
                          57            6A  D0 00066          MOVL     HDR2, R7                                        ; 0708
0050  8F              20  6E            00  2C 00069          MOVC5    #0, (SP), #32, #80, (R7)                        :
                                        67     00070                                                                   :

0050  8F              00  6E            00  2C 00071          MOVC5    #0, (SP), #0, #80, @HDR3                        ; 0709
```

```
                                              0000G  DF       00078                                                        ;
                                        56    0000G  CF  D0   0007B        MOVL     HDR4, R6                               ; 0710
    0050   8F           20              6E           00  2C   00080        MOVC5    #0, (SP), #32, #80, (R6)               ;
                                                     66       00087                                                        ;
                                        50       34  AB  D0   00088        MOVL     52(CURRENT_VCB), MVL                   ; 0714
                                        03       22  A0  91   0008C        CMPB     34(MVL), #3                            ; 0715
                                                     05  1B   00090        BLEQU    3$                                     ;
                                                 04  A6  94   00092        CLRB     4(R6)                                  ; 0717
                                                     06  11   00095        BRB      4$                                     ;
                                    43  A6     3030  8F  B0   00097  3$:   MOVW     #12336, 67(R6)                         ; 0719
                                    04  A7       46  8F  90   0009D  4$:   MOVB     #70, 4(R7)                             ; 0721
                                        6E           05  D0   000A2        MOVL     #5, DESCR                              ; 0722
                                    04  AE       05  A7  9E   000A5        MOVAB    5(R7), DESCR+4                         ; 0723
                                        50    0000G  CF  D0   000AA        MOVL     CURRENT_UCB, R0                        ; 0725
                                        7E       42  A0  3C   000AF        MOVZWL   66(R0), -(SP)                          ;
                                                 04  AE  9F   000B3        PUSHAB   DESCR                                  ;
                                                     7E  D4   000B6        CLRL     -(SP)                                  ;
                                               FF34  CF  9F   000B8        PUSHAB   CVT5                                   ;
                             00000000G  00           04  FB   000BC        CALLS    #4, SYS$FAO                            ;
                                        0A           50  E8   000C3        BLBS     R0, 5$                                 ;
                                        50           6A  D0   000C6        MOVL     HDR2, R0                               ; 0727
                      05   A0       FF2A  CF           05  28   000C9        MOVC3    #5, DEFAULT, 5(R0)                     ;
                                        57           6A  D0   000D0  5$:   MOVL     HDR2, R7                               ; 0729
                      0A   A7         05  A7           05  28   000D3        MOVC3    #5, 5(R7), 10(R7)                      ;
                                                 50  AB  B5   000D9        TSTW     80(CURRENT_VCB)                        ; 0731
                                                     28  13   000DC        BEQL     6$                                     ;
                                        6E           05  D0   000DE        MOVL     #5, DESCR                              ; 0734
                                    04  AE       0A  A7  9E   000E1        MOVAB    10(R7), DESCR+4                        ; 0735
                                        7E       50  AB  3C   000E6        MOVZWL   80(CURRENT_VCB), -(SP)                 ; 0737
                                                 04  AE  9F   000EA        PUSHAB   DESCR                                  ;
                                                     7E  D4   000ED        CLRL     -(SP)                                  ;
                                               FEFD  CF  9F   000EF        PUSHAB   CVT5                                   ;
                             00000000G  00           04  FB   000F3        CALLS    #4, SYS$FAO                            ;
                                        09           50  E8   000FA        BLBS     R0, 6$                                 ;
                                        50           6A  D0   000FD        MOVL     HDR2, R0                               ; 0739
                      0A   A0         05  A0           05  28   00100        MOVC3    #5, 5(R0), 10(R0)                      ;
                                        50           6A  D0   00106  6$:   MOVL     HDR2, R0                               ; 0746
                                    32  A0     3030  8F  B0   00109        MOVW     #12336, 50(R0)                         ;
                           57      0000G  CF 00000140  8F  C1   0010F        ADDL3    #320, HDR1, SCRATCH                    ; 0751
                                        7E       50  8F  9A   00119        MOVZBL   #80, -(SP)                             ; 0756
                                                     57  DD   0011D        PUSHL    SCRATCH                                ;
                                 0000G  CF           02  FB   0011F        CALLS    #2, READ_BLOCK                         ;
                                        5B           50  E9   00124        BLBC     R0, 8$                                 ;
                              32524448  8F           67  D1   00127        CMPL     (SCRATCH), #844252232                  ; 0759
                                                     52  12   0012E        BNEQ     8$                                     ;
                      00   BA               67     0050  8F  28   00130        MOVC3    #80, (SCRATCH), @HDR2                  ; 0762
                                        58           02  D0   00137        MOVL     #2, NUMBER_OF_LABELS                   ; 0763
                                        44       2D  AB  E9   0013A        BLBC     45(CURRENT_VCB), 8$                    ; 0765
                                        7E       50  8F  9A   0013E        MOVZBL   #80, -(SP)                             ; 0768
                                                     57  DD   00142        PUSHL    SCRATCH                                ;
                                 0000G  CF           02  FB   00144        CALLS    #2, READ_BLOCK                         ;
                                        36           50  E9   00149        BLBC     R0, 8$                                 ;
                              33524448  8F           67  D1   0014C        CMPL     (SCRATCH), #861029448                  ; 0771
                                                     0B  12   00153        BNEQ     7$                                     ;
                   0000G   DF               67     0050  8F  28   00155        MOVC3    #80, (SCRATCH), @HDR3                  ; 0774
                                        58           03  D0   0015D        MOVL     #3, NUMBER_OF_LABELS                   ; 0775
                                        7E       50  8F  9A   00160  7$:   MOVZBL   #80, -(SP)                             ; 0777
```

```
                                            57  DD 00164          PUSHL   SCRATCH                                   ;
                         0000G  CF          02  FB 00166          CALLS   #2, READ_BLOCK                            ;
                                14          50  E9 0016B          BLBC    R0, 8$                                    ;
                      34524448  8F          67  D1 0016E          CMPL    (SCRATCH), #877806664                     ; 0779
                                            0B  12 00175          BNEQ    8$                                        ;
               0000G  DF        67    0050  8F  28 00177          MOVC3   #80, (SCRATCH), @HDR4                     ; 0782
                                58          04  D0 0017F          MOVL    #4, NUMBER_OF_LABELS                      ; 0783
                         0000G  CF          00  FB 00182 8$:      CALLS   #0, CHCK_IO_CLR_EXCP                      ; 0792
                                            58  DD 00187          PUSHL   NUMBER_OF_LABELS                          ; 0793
                                            01  DD 00189          PUSHL   #1                                        ;
                                            5E  DD 0018B          PUSHL   SP                                        ;
                                      0000V CF  9F 0018D          PUSHAB  SET_NUMBER_OF_LABELS                      ;
                                69          04  FB 00191          CALLS   #4, SYS$CMKRNL                            ;
                                50          01  D0 00194          MOVL    #1, R0                                    ; 0794
                                                04 00197          RET                                               ;
                                            50  D4 00198 9$:      CLRL    R0                                        ; 0796
                                                04 0019A          RET                                               ;

; Routine Size:  411 bytes,    Routine Base:  $CODE$ + 00A4


;  416          0797  1
```

```
1 GLOBAL ROUTINE WRAP_AROUND : L$WRAP_AROUND =
1
1 !++
1 !
1 ! FUNCTIONAL DESCRIPTION:
1 !       If this is not the first time through and the search started
1 !       at the beginning of the volume set then return error else rewind volume set
1 !
1 ! CALLING SEQUENCE:
1 !       WRAP_AROUND()
1 !
1 ! INPUT PARAMETERS:
1 !       none
1 !
1 ! IMPLICIT INPUTS:
1 !       LOCAL_FIB - copy of user's fib
1 !       CURRENT_VCB - address of current volume VCB
1 !
1 ! OUTPUT PARAMETERS:
1 !       none
1 !
1 ! IMPLICIT OUTPUTS:
1 !       none
1 !
1 ! ROUTINE VALUE:
1 !       0 back to beginning of search
1 !       1 at beginning of volume set
1 !
1 ! SIDE EFFECTS:
1 !       none
1 !
1 !--
1
2     BEGIN
2
2     EXTERNAL REGISTER
2         COMMON_REG;
2
2     EXTERNAL ROUTINE
2         MOUNT_VOL        : COMMON_CALL,           ! mount volume
2         REWIND_VOL_SET   : COMMON_CALL;           ! rewind volume set
2
2     EXTERNAL
2         LOCAL_FIB        : BBLOC                  ! copy of user's fib
2
2     IF .CURRENT_VCB[VCB$L_START_FID] EQL %X'00010001'
2     THEN
2         RETURN 0
2     ELSE
3         BEGIN
3         REWIND_VOL_SET();
3
3         ! get first volume mounted
3         !
3         MOUNT_VOL(1, $FIELDMASK(MOU$V_REWIND) + $FIELDMASK(MOU$V_LBLCHECK));
3
3         IF NOT READ_HDR()
```

```
;  475          0855  3         THEN
;  476          0856  3             ERR_EXIT(SS$_TAPEPOSLOST);
;  477          0857  3
;  478          0858  2         END;
;  479          0859  2
;  480          0860  2     RETURN 1;
;  481          0861  2
;  482          0862  1     END;                                        ! end of routine


                                                                .EXTRN  REWIND_VOL_SET

                            00010001   8F        28    AB   D1 00000 WRAP_AROUND::
                                                                CMPL    40(CURRENT_VCB), #65537                   ; 0843
                                                       1E   13 00008        BEQL    2$                                        ;
                                0000G  CF              00   FB 0000A        CALLS   #0, REWIND_VOL_SET                        ; 0848
                                                       03   DD 0000F        PUSHL   #3                                        ; 0852
                                                       01   DD 00011        PUSHL   #1                                        ;
                                0000G  CF              02   FB 00013        CALLS   #2, MOUNT_VOL                             ;
                                FE48   CF              00   FB 00018        CALLS   #0, READ_HDR                              ; 0854
                                       04              50   E8 0001D        BLBS    R0, 1$                                    ;
                                             0224      8F   BF 00020        CHMU    #548                                      ; 0856
                                       50              01   D0 00024 1$:    MOVL    #1, R0                                    ; 0860
                                                            05 00027        RSB                                               ;
                                                       50   D4 00028 2$:    CLRL    R0                                        ; 0862
                                                            05 0002A        RSB                                               ;

; Routine Size:  43 bytes,    Routine Base:  $CODE$ + 023F

;  483          0863  1
```

```
1 GLOBAL ROUTINE SPACE_EOF : COMMON_CALL NOVALUE =
1
1 !++
1 !
1 ! FUNCTIONAL DESCRIPTION:
1 !       This routine spaces to the end of the current file, right
1 !       before the next file.
1 !
1 ! CALLING SEQUENCE:
1 !       SPACE_EOF()
1 !
1 ! INPUT PARAMETERS:
1 !       none
1 !
1 ! IMPLICIT INPUTS:
1 !       CURRENT_VCB _ address of current VCB
1 !
1 ! OUTPUT PARAMETERS:
1 !       none
1 !
1 ! IMPLICIT OUTPUTS:
1 !       none
1 !
1 ! ROUTINE VALUE:
1 !       none
1 !
1 ! SIDE EFFECTS:
1 !       The tape is left positioned in front of HDR1 of the next file
1 !
1 !--
1
2    BEGIN
2
2    SWITCHES NOOPTIMIZE;
2
2    EXTERNAL REGISTER
2        COMMON_REG;
2
2    EXTERNAL ROUTINE
2        GTNEXT_VOL_READ : JSB,                ! get next volume on read
2        READ_BLOCK      : COMMON_CALL,        ! read mag tape block
2        SPACE_TM        : COMMON_CALL;        ! space tm's
2
2    EXTERNAL
2        CURRENT_UCB     : REF BBLOCK;         ! address of current ucb
2
2    LOCAL
2        TM;
2
2    ! If tape is positioned in header set, space 2 tape marks
2    !
2
2    IF .CURRENT_VCB[VCB$B_TM] EQL 0 AND .HDR1[HD1$L_HD1LID] EQL 'HDR1'
2    THEN
2        SPACE_TM(2);
2
2    ! if in data area, space 1 tape mark
```

```
;  542      0921  2    !
;  543      0922  2
;  544      0923  2    IF .CURRENT_VCB[VCB$B_TM] EQLU 1
;  545      0924  2    THEN
;  546      0925  2        SPACE_TM(1);
;  547      0926  2
;  548      0927  2    ! Now if trailer label has not been read, read it
;  549      0928  2    !
;  550      0929  2
;  551      0930  2    IF .CURRENT_VCB[VCB$B_TM] EQLU 2
;  552      0931  2        AND
;  553      0932  2        (.CURRENT_UCB[UCB$L_RECORD] - .CURRENT_VCB[VCB$L_ST_RECORD]) EQL 0
;  554      0933  2    THEN
;  555      0934  2
;  556      0935  2        IF NOT READ_BLOCK(.HDR1, ANSI_LBLSZ)
;  557      0936  2        THEN
;  558      0937  2            ERR_EXIT(SS$_TAPEPOSLOST);
;  559      0938  2
;  560      0939  2    WHILE 1
;  561      0940  2    DO
;  562      0941  3        BEGIN
;  563      0942  3
;  564      0943  3        IF .HDR1[HD1$L_HD1LID] EQL 'EOF1'
;  565      0944  3        THEN
;  566      0945  3            EXITLOOP;
;  567      0946  3
;  568      0947  3        IF .HDR1[HD1$L_HD1LID] NEQ 'EOV1'
;  569      0948  3        THEN
;  570      0949  3            ERR_EXIT(SS$_TAPEPOSLOST);
;  571      0950  3
;  572      0951  3        GTNEXT_VOL_READ();
;  573      0952  3
;  574      0953  3        IF .CURRENT_VCB[VCB$B_TM] EQLU 0
;  575      0954  3        THEN
;  576      0955  3            SPACE_TM(2)
;  577      0956  3        ELSE
;  578      0957  3            SPACE_TM(1);
;  579      0958  3
;  580      0959  3        IF NOT READ_BLOCK(.HDR1, ANSI_LBLSZ)
;  581      0960  3        THEN
;  582      0961  3            ERR_EXIT(SS$_TAPEPOSLOST);
;  583      0962  3
;  584      0963  2        END;
;  585      0964  2
;  586      0965  2    IF .CURRENT_VCB[VCB$B_TM] EQLU 2
;  587      0966  2    THEN
;  588      0967  2        SPACE_TM(1);
;  589      0968  2
;  590      0969  1    END;                                          ! end of routine


                                                                   .EXTRN  GTNEXT_VOL_READ
                                                                   .EXTRN  SPACE_TM

                                            07FC 00000             .ENTRY  SPACE_EOF, Save R2,R3,R4,R5,R6,R7,R8,R9,R10 ; 0864
                              52    0000G  CF   9E 00002           MOVAB   SPACE_TM, R2                                ;
```

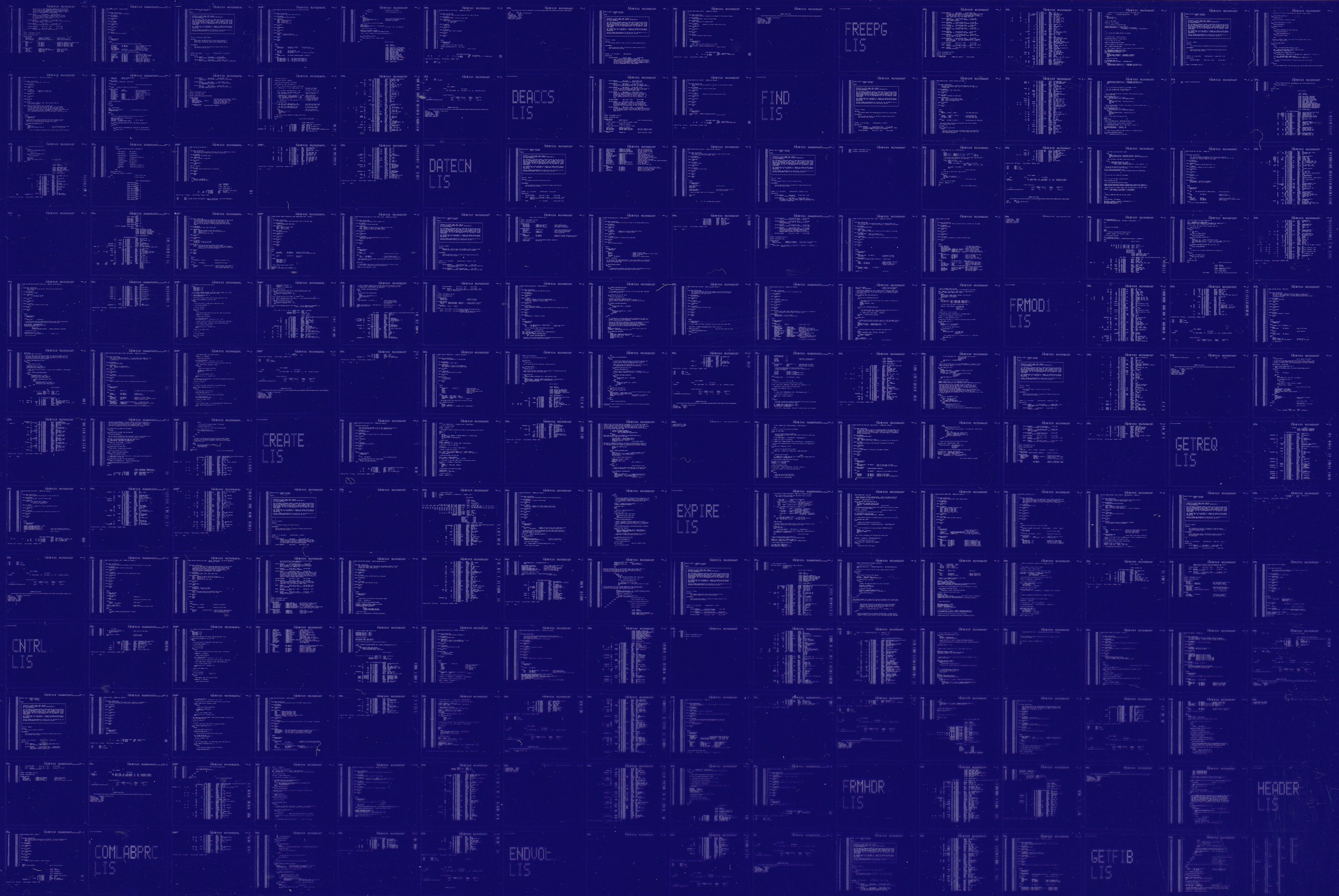
```
                             2E    AB   95 00007             TSTB    46(CURRENT_VCB)                        ; 0916
                                   10   12 0000A             BNEQ    1$                                     ;
  31524448    8F          0000G    DF   D1 0000C             CMPL    @HDR1, #827475016                      ;
                                   05   12 00015             BNEQ    1$                                     ;
                                   02   DD 00017             PUSHL   #2                                     ; 0918
              62                   01   FB 00019             CALLS   #1, SPACE_TM                           ;
              01             2E    AB   91 0001C  1$:        CMPB    46(CURRENT_VCB), #1                    ; 0923
                                   05   12 00020             BNEQ    2$                                     ;
                                   01   DD 00022             PUSHL   #1                                     ; 0925
              62                   01   FB 00024             CALLS   #1, SPACE_TM                           ;
              02             2E    AB   91 00027  2$:        CMPB    46(CURRENT_VCB), #2                    ; 0930
                                   21   12 0002B             BNEQ    4$                                     ;
              50          0000G    CF   D0 0002D             MOVL    CURRENT_UCB, R0                        ; 0932
        30    AB          00B0     C0   D1 00032             CMPL    176(R0), 48(CURRENT_VCB)               ;
                                   14   12 00038             BNEQ    4$                                     ;
              7E             50    8F   9A 0003A  3$:        MOVZBL  #80, -(SP)                             ; 0935
                          0000G    CF   DD 0003E             PUSHL   HDR1                                   ;
     0000G    CF                   02   FB 00042             CALLS   #2, READ_BLOCK                         ;
              04                   50   E8 00047             BLBS    R0, 4$                                 ;
                           0224    8F   BF 0004A             CHMU    #548                                   ; 0937
  31464F45    8F          0000G    DF   D1 0004E  4$:        CMPL    @HDR1, #826691397                      ; 0943
                                   22   13 00057             BEQL    8$                                     ;
  31564F45    8F          0000G    DF   D1 00059             CMPL    @HDR1, #827739973                      ; 0947
                                   04   13 00062             BEQL    5$                                     ;
                           0224    8F   BF 00064             CHMU    #548                                   ; 0949
                                0000G   30 00068  5$:        BSBW    GTNEXT_VOL_READ                        ; 0951
                             2E    AB   95 0006B             TSTB    46(CURRENT_VCB)                        ; 0953
                                   04   12 0006E             BNEQ    6$                                     ;
                                   02   DD 00070             PUSHL   #2                                     ; 0955
                                   02   11 00072             BRB     7$                                     ;
                                   01   DD 00074  6$:        PUSHL   #1                                     ; 0957
              62                   01   FB 00076  7$:        CALLS   #1, SPACE_TM                           ;
                                   BF   11 00079             BRB     3$                                     ; 0959
              02             2E    AB   91 0007B  8$:        CMPB    46(CURRENT_VCB), #2                    ; 0965
                                   05   12 0007F             BNEQ    9$                                     ;
                                   01   DD 00081             PUSHL   #1                                     ; 0967
              62                   01   FB 00083             CALLS   #1, SPACE_TM                           ;
                                        04 00086  9$:        RET                                            ; 0969

; Routine Size:  135 bytes,    Routine Base:  $CODE$ + 026A

;  591          0970  1
```

```
1 ROUTINE MAKE_CUR_FILE (LABELS) : COMMON_CALL NOVALUE =
1
1 !++
1 !
1 ! FUNCTIONAL DESCRIPTION:
1 !       This routine updates the current file number and the Starlet
1 !       file indicator.
1 !
1 ! CALLING SEQUENCE:
1 !       MAKE_CUR_FILE(ARG1), call in kernel mode
1 !
1 ! INPUT PARAMETERS:
1 !       ARG1 - address of labels
1 !
1 ! IMPLICIT INPUTS:
1 !       none
1 !
1 ! OUTPUT PARAMETERS:
1 !       none
1 !
1 ! IMPLICIT OUTPUTS:
1 !       If file is Starlet file, then VCB$V_STARFILE = 1
1 !       CUR_NUM is updated
1 !
1 ! ROUTINE VALUE:
1 !       none
1 !
1 ! SIDE EFFECTS:
1 !       none
1 !
1 !--
1
2    BEGIN
2
2    EXTERNAL REGISTER
2        COMMON_REG;
2
2    MAP
2        LABELS  : REF BBLOCK;                    ! HDR1, HDR2, and HDR3 address
2
2    BIND
2
2        ! Any file with 11 code will be supported, instead of only 11A
2        !
2        STARID = UPLIT ('DECFILE11');
2
2    EXTERNAL ROUTINE
2        FORMAT_FID      : COMMON_CALL;           ! format file id
2
2    CURRENT_VCB[VCB$V_STARFILE] = CH$EQL(9, STARID, 9, LABELS[HD1$T_SYSCODE],0);
2    FORMAT_FID(CURRENT_VCB[VCB$W_CUR_NUM]);
1    END;                                         ! end of routine


                                                  002F1          .BLKB    3
 00  00  00  31  31  45  4C  49  46  43  45  44   002F4 P.AAD:   .ASCII   \DECFILE11\<0><0><0>              ;
```

```
                                                          STARID=           P.AAD
                                                                  .EXTRN  FORMAT_FID

                                               001C 00000 MAKE_CUR_FILE:
                                                                  .WORD   Save R2,R3,R4                        ; 0971
                                    50     04  AC  D0 00002       MOVL    LABELS, R0                           ; 1020
                                               54  D4 00006       CLRL    R4                                   ;
                  3C   A0      E8   AF         09  29 00008       CMPC3   #9, STARID, 60(R0)                   ;
                                               02  12 0000E       BNEQ    1$                                   ;
                                               54  D6 00010       INCL    R4                                   ;
   2D   AB            01            00         54  F0 00012 1$:   INSV    R4, #0, #1, 45(CURRENT_VCB)          ;
                                           24  AB  9F 00018       PUSHAB  36(CURRENT_VCB)                      ; 1021
                               0000G  CF       01  FB 0001B       CALLS   #1, FORMAT_FID                       ;
                                                   04 00020       RET                                          ; 1022

; Routine Size:  33 bytes,    Routine Base:  $CODE$ + 0300


;  645          1023  1
;  647          1024  1 GLOBAL ROUTINE UPDVCB_LEOV (BIT_VALUE) : COMMON_CALL NOVALUE =
;  648          1025  1
;  649          1026  1 !++
;  650          1027  1 !
;  651          1028  1 ! FUNCTIONAL DESCRIPTION:
;  652          1029  1 !       This routine sets or clears the logical end of file bit in the VCB
;  653          1030  1 !
;  654          1031  1 ! CALLING SEQUENCE:
;  655          1032  1 !       UPDVCB_LEOV(ARG1), called in kernel mode
;  656          1033  1 !
;  657          1034  1 ! INPUT PARAMETERS:
;  658          1035  1 !       value to set logical end of volume to:
;  659          1036  1 !               0 - clear bit
;  660          1037  1 !               1 - set bit
;  661          1038  1 !
;  662          1039  1 ! IMPLICIT INPUTS:
;  663          1040  1 !       CURRENT_VCB - address of volume control block
;  664          1041  1 !
;  665          1042  1 ! OUTPUT PARAMETERS:
;  666          1043  1 !       none
;  667          1044  1 !
;  668          1045  1 ! IMPLICIT OUTPUTS:
;  669          1046  1 !       CURRENT_VCB[VCB$V_LOGICEOVS] is set or cleared
;  670          1047  1 !
;  671          1048  1 ! ROUTINE VALUE:
;  672          1049  1 !       none
;  673          1050  1 !
;  674          1051  1 ! SIDE EFFECTS:
;  675          1052  1 !       none
;  676          1053  1 !
;  677          1054  1 !--
;  678          1055  1
;  679          1056  2     BEGIN
;  680          1057  2
;  681          1058  2     EXTERNAL REGISTER
;  682          1059  2         COMMON_REG;
;  683          1060  2
;  684          1061  2     CURRENT_VCB[VCB$V_LOGICEOVS] = .BIT_VALUE;
;  685          1062  1     END;                                            ! end of routine



                                                        0000 00000         .ENTRY  UPDVCB_LEOV, Save nothing                    ; 1024
     0B   AB            01            01        04   AC   F0 00002         INSV    BIT_VALUE, #1, #1, 11(CURRENT_VCB)           ; 1061
                                                          04 00009         RET                                                  ; 1062

; Routine Size:  10 bytes,    Routine Base:  $CODE$ + 0321


;  686          1063  1
;  688          1064  1 ROUTINE SET_NUMBER_OF_LABELS (NUMBER_OF_LABELS) : COMMON_CALL NOVALUE =
;  689          1065  1
;  690          1066  1 !++
;  691          1067  1 !
;  692          1068  1 ! FUNCTIONAL DESCRIPTION:
;  693          1069  1 !       This routine sets then number of labels read by the MTAACP in the VCB.
;  694          1070  1 !       This value will be used to determine how many labels are written out
;  695          1071  1 !       won volume switch or at end of file  `ocessing. The reason this is
;  696          1072  1 !       necessary is so that if a file is o    with fewer labels then we support
;  697          1073  1 !       we do not write the greater number     DR labels out to the tape. This
;  698          1074  1 !       would be a noncompliance with the ANS. standard for tape label
;  699          1075  1 !       processing.
;  700          1076  1 !
;  701          1077  1 ! CALLING SEQUENCE:
;  702          1078  1 !       SET_NUMBER_OF_LABELS(ARG1), called in kernel mode
;  703          1079  1 !
;  704          1080  1 ! INPUT PARAMETERS:
;  705          1081  1 !       Number of labels read.
;  706          1082  1 !
;  707          1083  1 ! IMPLICIT INPUTS.
;  708          1084  1 !       CURRENT_VCB - address of volume control block
;  709          1085  1 !
;  710          1086  1 ! OUTPUT PARAMETERS:
;  711          1087  1 !       none
;  712          1088  1 !
;  713          1089  1 ! IMPLICIT OUTPUTS:
;  714          1090  1 !       CURRENT_VCB[VCB$B_LBLC  ] is set
;  715          1091  1 !
;  716          1092  1 ! ROUTINE VALUE:
;  717          1093  1 !       none
;  718          1094  1 !
;  719          1095  1 ! SIDE EFFECTS:
;  720          1096  1 !       none
;  721          1097  1 !
;  722          1098  1 !--
;  723          1099  1
;  724          1100  2     BEGIN
;  725          1101  2
;  726          1102  2     EXTERNAL REGISTER
;  727          1103  2         COMMON_REG;
;  728          1104  2
;  729          1105  2     CURRENT_VCB[VCB$B_LBLCNT] = .NUMBER_OF_LABELS;
;  730          1106  1     END;                                        ! end of routine



                                          0000 00000 SET_NUMBER_OF_LABELS:
                                                               .WORD   Save nothing                                 ; 1064
                          48   AB        04   AC  90 00002     MOVB    NUMBER_OF_LABELS, 72(CURRENT_VCB)            ; 1105
                                                  04 00007     RET                                                  ; 1106

; Routine Size:  8 bytes,    Routine Base:  $CODE$ + 032B


;  731          1107  1
;  732          1108  1 END
;  733          1109  1
;  734          1110  0 ELUDOM




;                                       PSECT SUMMARY

;       Name                         Bytes                       Attributes

;  $CODE$                                 819  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)




;                              Library Statistics

;                                            -------- Symbols --------      Pages       Processing
;       File                                 Total    Loaded   Percent      Mapped      Time

;  _$255$DUA28:[SYSLIB]LIB.L32;1             18619        36         0      1000          00:01.8





;                                       COMMAND QUALIFIERS

;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS$:HEADER/OBJ=OBJ$:HEADER MSRC$:HEADER/UPDATE=(ENH$:HEADER)

; Size:          782 code + 37 data bytes
; Run Time:         00:17.7
; Elapsed Time:     00:40.6
; Lines/CPU Min:      3771
; Lexemes/CPU-Min: 18091
; Memory Used:  163 pages
; Compilation Complete
```

0254 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY



0255 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY