K 6

```
FFFFFFFFFFFFFFF      000000000      RRRRRRRRRRRR      RRRRRRRRRRRR      TTTTTTTTTTTTTTT   LLL
FFFFFFFFFFFFFFF      000000000      RRRRRRRRRRRR      RRRRRRRRRRRR      TTTTTTTTTTTTTTT   LLL
FFFFFFFFFFFFFFF      000000000      RRRRRRRRRRRR      RRRRRRRRRRRR      TTTTTTTTTTTTTTT   LLL
FFF               000         000   RRR         RRR   RRR         RRR         TTT         LLL
FFF               000         000   RRR         RRR   RRR         RRR         TTT         LLL
FFF               000         000   RRR         RRR   RRR         RRR         TTT         LLL
FFF               000         000   RRR         RRR   RRR         RRR         TTT         LLL
FFF               000         000   RRR         RRR   RRR         RRR         TTT         LLL
FFF               000         000   RRR         RRR   RRR         RRR         TTT         LLL
FFFFFFFFFFFF      000         000   RRRRRRRRRRRR      RRRRRRRRRRRR            TTT         LLL
FFFFFFFFFFFF      000         000   RRRRRRRRRRRR      RRRRRRRRRRRR            TTT         LLL
FFFFFFFFFFFF      000         000   RRRRRRRRRRRR      RRRRRRRRRRRR            TTT         LLL
FFF               000         000   RRR   RRR         RRR   RRR               TTT         LLL
FFF               000         000   RRR   RRR         RRR   RRR               TTT         LLL
FFF               000         000   RRR   RRR         RRR   RRR               TTT         LLL
FFF               000         000   RRR      RRR      RRR      RRR            TTT         LLL
FFF               000         000   RRR      RRR      RRR      RRR            TTT         LLL
FFF               000         000   RRR      RRR      RRR      RRR            TTT         LLL
FFF                  000000000      RRR         RRR   RRR         RRR         TTT         LLLLLLLLLLLLLLL
FFF                  000000000      RRR         RRR   RRR         RRR         TTT         LLLLLLLLLLLLLLL
FFF                  000000000      RRR         RRR   RRR         RRR         TTT         LLLLLLLLLLLLLLL
```

```
**FILE**ID**FOROPENDE                          L  8


    FFFFFFFFFF    OOOOOO    RRRRRRRR     OOOOOO    PPPPPPPP     EEEEEEEEEE  NN      NN  DDDDDDDD     EEEEEEEEEE
    FFFFFFFFFF    OOOOOO    RRRRRRRR     OOOOOO    PPPPPPPP     EEEEEEEEEE  NN      NN  DDDDDDDD     EEEEEEEEEE
    FF          OO      OO  RR      RR  OO      OO  PP      PP  EE          NN      NN  DD      DD   EE
    FF          OO      OO  RR      RR  OO      OO  PP      PP  EE          NN      NN  DD      DD   EE
    FF          OO      OO  RR      RR  OO      OO  PP      PP  EE          NNNN    NN  DD      DD   EE
    FF          OO      OO  RR      RR  OO      OO  PP      PP  EE          NNNN    NN  DD      DD   EE
    FFFFFFFF    OO      OO  RRRRRRRR    OO      OO  PPPPPPPP    EEEEEEEE    NN  NN  NN  DD      DD   EEEEEEEE
    FFFFFFFF    OO      OO  RRRRRRRR    OO      OO  PPPPPPPP    EEEEEEEE    NN  NN  NN  DD      DD   EEEEEEEE
    FF          OO      OO  RR  RR      OO      OO  PP          EE          NN    NNNN  DD      DD   EE
    FF          OO      OO  RR  RR      OO      OO  PP          EE          NN    NNNN  DD      DD   EE
    FF          OO      OO  RR    RR    OO      OO  PP          EE          NN      NN  DD      DD   EE
    FF          OO      OO  RR    RR    OO      OO  PP          EE          NN      NN  DD      DD   EE          ....
    FF            OOOOOO    RR      RR    OOOOOO    PP          EEEEEEEEEE  NN      NN  DDDDDDDD     EEEEEEEEEE  ....
    FF            OOOOOO    RR      RR    OOOOOO    PP          EEEEEEEEEE  NN      NN  DDDDDDDD     EEEEEEEEEE  ....
                                                                                                                ....

    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 MODULE FOR$$OPEN_DEFLT (%TITLE 'FORTRAN default open'
0                  IDENT = '1-098'           ! File: FOROPENDE.B32  Edit: LEB1098
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 ! FACILITY: FORTRAN Support Library - not user callable
1 !
1 ! ABSTRACT:
1 !
1 !       This module contains a routine to perform default file
1 !       opening for FORTRAN programs.
1 !
1 ! ENVIRONMENT: User access mode; mixture of AST level or not.
1 !
1 ! AUTHOR:        Thomas N. Hastings, CREATION DATE: 6-Mar-77; Version 0
1 !
1 ! MODIFIED BY:
1 !
1 !       Thomas N. Hastings, 15-Mar-77: Version 0
1 ! [Previous edit history removed.  SBL 5-Oct-1982]
1 ! 1-078 - Add support for DEFAULTFILE=string.  JAW 30-Jun-1981
1 ! 1-079 - Increase default value of RECL for unformatted variable-length
1 !         records from 126 to 2046, to improve performance when
1 !         RECORDTYPE='SEGMENTED'.  JAW 17-Jul-1981
1 ! 1-080 - Fix logic error in record type check made when user does not
1 !         specify record type for an old file.  (Allowed both FIXED and
1 !         SEGMENTED to be set simultaneously.)  JAW 25-Aug-1981
1 ! 1-081 - Change algorithm for determining the length of a list-directed
1 !         output record: use RECL if specified, else 80/81 depending on
1 !         carriage control.  JAW 26-Aug-1981
1 ! 1-082 - Add test for blocksize less than recordsize (made only if open
1 !         or create fails and device is mag tape).  If so, signal
1 !         INCRECLEN since RMS does not give a useful message in this
```

```
;      1 !            case.  JAW 28-Aug-1981
;      1 ! 1-083 - Save and restore the STS and STV around the $PARSE we do if we
;      1 !            get an unexpected error.  SBL 28-Sep-1981
;      1 ! 1-084 - Signal FOR$K_OPEFAI if RMS$_WLK and not readonly.  DGP 03-Dec-1981
;      1 ! 1-085 - Set the MRS in the FAB for indexed files.  DGP 21-Dec-1981
;      1 ! 1-086 - Allow existing file to be SEGMENTED only if it has RFM=VAR.
;      1 !            Correct 1-082 and 1-084 so that only RMS$_CRE errors check for
;      1 !            INCRECLEN.  SBL 13-Jan-1982
;      1 ! 1-087 - Complete 1-085.  It was much too simplistic and caused existing ISAM
;      1 !            files to not be able to opened.  DGP 22-Feb-1982
;      1 ! 1-088 - Unfortunately, 1-087 did not allow existing ISAM files with an MRS
;      1 !            smaller than the default buffer size to be opened unless the
;      1 !            RECL was explicitly specified.  Fix it.  SBL 16-Apr-1982
;      1 ! 1-089 - For devices other than disks and terminals, reduce the default
;      1 !            recordsize to less than the blocksize, if necessary.  Use blocksize
;      1 !            as recordsize on existing files, if no MRS or LRL.  SBL 30-Sep-1982
;      1 ! 1-090 - Make default unformmated RECL 2044 instead of 2046.  This allows
;      1 !            default disk files to be copied to tape.  SBL 8-Nov-1982
;      1 ! 1-091 - Reflect change that OTS$$ data structures are now FOR$$.  SBL 8-Nov-1982
;      1 ! 1-092 - Restore some INCOPECLO checks that were mistakenly deleted
;      1 !            in an earlier edit.  Use new macro to call FOR$$SIGNAL_STO.
;      1 !            Move FAB and NAM to heap at end of RAB.  Add support for stream
;      1 !            recordtypes.  Raise bucketsize limit to 63.  Use carriagecontrol
;      1 !            specified/defaulted if PPF.  Don't decrement recordlength by 4
;      1 !            unless SEGMENTED.  SBL 29-Mar-1983
;      1 ! 1-093 - Add RFA cacheing for BACKSPACE.  SBL 2-June-1983
;      1 ! 1-094 - Restrict stream recordtypes to sequential org only.  SBL 28-Jul-1983
;      1 ! 1-095 - Use LNM$C_NAMLENGTH for maximum size of equivalence string in call
;      1 !            to $TRNLOG.  DG 8-Nov-1983
;      1 ! 1-096 - Add stack location of TEMP_FNS to store the temporary filespec
;      1 !            for ASSIGN.  Also change back use of LNM$C_NAMLENGTH to be
;      1 !            NAM$C_MAXRSS.  LEB 2-Feb-1984
;      1 ! 1-097 - Free KEY_XABs when an open fails. STAN 27-Feb-1984.
;      1 ! 1-098 - Disassociate the NAM block during the $PARSE to clear up a
;      1 !            problem associated with floating memory.  LEB 21-Mar-1984
;      1 !--
;      1
```

```
1 !
1 ! PROLOGUE FILE:
1 !
1
1 REQUIRE 'RTLIN:FORPROLOG';                                  ! FORTRAN definitions
1
1 !
1 ! TABLE OF CONTENTS:
1 !
1
1 FORWARD ROUTINE
1     FOR$$OPEN_DEFLT : CALL_CCB NOVALUE,                     ! default OPEN
1     FOR$$OPEN_PROC : CALL_CCB NOVALUE;                      ! common OPEN procedure
1
1 !
1 ! MACROS:
1 !
1 !       NONE
1 !
1 ! EQUATED SYMBOLS:
1 !
1 !       NONE
1 !
1 ! OWN STORAGE:
1 !
1 !       NONE
1 !
1 ! EXTERNAL REFERENCES:
1 !
1
1 EXTERNAL ROUTINE
1     FOR$$ERR_OPECLO,                                        ! OPEN/CLOSE condition handler
1     FOR$$SIGNAL_STO : NOVALUE,                              ! Convert small FORTRAN err #
1                                                             ! to 32-bit VAX error # and SIGNAL_STOP
1     FOR$$SIG_NO_LUB : NOVALUE,                              ! same as FOR$$SIGNAL_STO except no LUB setup
1                                                             ! so must pass LUN explicitly.
1     FOR$$CB_PUSH : JSB_CB_PUSH NOVALUE,                     ! push current LUB/ISB/RAB, if any, and allocate LUB/ISB/RAB
1                                                             ! for this logical unit
1     FOR$$CB_POP : JSB_CB_POP NOVALUE,                       ! Pop I/O system back to previous LUB or indicate
1                                                             ! no I/O statement is currently being processed.
1     FOR$$GET_VM,                                            ! Allocate virtual memory
1     FOR$$FREE_VM : NOVALUE,                                 ! Free virtual memory
1     FOR$$SIG_FATINT : NOVALUE,                              ! Signal_stop internal error
1     FOR$$DECL_EXITH : NOVALUE;                              ! Declare the exit handler
1
1 EXTERNAL
1     FOR$$L_XIT_LOCK;                                        ! True if exit handler already declared
1
```

```
1 GLOBAL ROUTINE FOR$$OPEN_DEFLT (                       ! Default OPEN
1           ACCESS_VAL,                                  ! Access = OPEN$K_ACC_{SEQ, DIR}
1           TYPE_VAL,                                    ! TYPE = OPEN$K_ACC_{NEW,OLD}
1           FORM_VAL)                                    ! FORM = OPEN$K_FOR_{UNF, FOR, UNS}
1     : CALL_CCB NOVALUE =
1
1 !++
1 ! ABSTRACT:
1 !
1 !         Perform default OPEN for an I/O statement for the indicated
1 !         logical unit. The possible parameters are a restricted
1 !         subset of explicit OPEN, plus FORM = 'UNSPECIFIED' (for
1 !         ENDFILE only). The keywords for default OPEN are:
1 !         ACCESS, TYPE, and FORM.
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !         LUB_ADR.mlu.ra            adr of LUB/ISB/RAB control block
1 !         ACCESS_VAL.rlu.v          Value = OPEN$K_ACC_{SEQ,DIR}
1 !                                   to indicate ACCESS = 'SEQUENTIAL'
1 !                                   or 'DIRECT'.
1 !         TYPE_VAL.rlu.v            Value = OPEN$K_TYPE_{NEW, OLD} TO
1 !                                   indicate TYPE = 'NEW' or 'OLD'
1 !         FORM_VAL.rlu.v            Value = OPEN$K_FORM_{UNF, FOR, UNS}
1 !                                   to indicate FORM = 'UNFORMATTED',
1 !                                   'FORMATTED', or 'UNSPECIFIED
1 !                                   (ENDFILE only).
1 !
1 ! IMPLICIT INPUTS:
1 !
1 !         LUB$V_READ_ONLY           1 if 'READONLY' specified in CALL FDBSET
1 !         LUB$V_DIRECT              1 if specified on previous DEFINEFILE
1 !         LUB$V_OLD_FILE            1 if specified on previous CALL FDBSET
1 !         LUB$V_UNFORMAT            1 if specified on previous DEFINEFILE
1 !         LUB$W_LUN                 FORTRAN logical unit number
1 !
1 ! IMPLICIT OUTPUTS:
1 !
1 !         LUB$V_DIRECT              1 if ACCESS = 'DIRECT' or DEFINEFILE
1 !         LUB$V_OLD_FILE            1 if TYPE = 'OLD' or CALL FDBSET 'OLD'
1 !         LUB$V_FORMATTED           1 if FORM = 'FORMATTED'
1 !         LUB$V_UNFORMAT            1 if FORM = 'UNFORMATTED' or DEFINEFILE
1 !
1 ! COMPLETION STATUS:
1 !
1 !         NONE
1 !
1 ! SIDE EFFECTS:
1 !
1 !         See FOR$$OPEN_PROC for SIGNAL_STOPs.
1 !--
1
2     BEGIN
2
2     EXTERNAL REGISTER
2         CCB : REF $FOR$CCB_DECL;
2
```

```
;  202          0265  2       LOCAL
;  203          0266  2           OPEN : VECTOR [OPEN$K_KEY_MAX + 1];       ! OPEN parameter array
;  204          0267  2
;  205          0268  2       !+
;  206          0269  2       ! Clear OPEN parameter array
;  207          0270  2       !-
;  208          0271  2
;  209          0272  2       CH$FILL (0, (OPEN$K_KEY_MAX + 1)*%UPVAL, OPEN);
;  210          0273  2
;  211          0274  2       !+
;  212          0275  2       ! Setup count, ACCESS, TYPE, and FORM parameter values
;  213          0276  2       !-
;  214          0277  2
;  215          0278  2       OPEN [OPEN$K_ACCESS] = .ACCESS_VAL;
;  216          0279  2       OPEN [OPEN$K_TYPE] = .TYPE_VAL;
;  217          0280  2       OPEN [OPEN$K_FORM] = .FORM_VAL;
;  218          0281  2
;  219          0282  2       !+
;  220          0283  2       ! Perform the OPEN - call common procedure with a pointer
;  221          0284  2       ! to the OPEN parameter VECTOR of longword values.
;  222          0285  2       !-
;  223          0286  2
;  224          0287  2       FOR$$OPEN_PROC (OPEN);
;  225          0288  2       RETURN;
;  226          0289  1       END;                                          ! End of FOR$OPEN_DEFLT routine


                                                                              .TITLE  FOR$$OPEN_DEFLT FORTRAN default open
                                                                              .IDENT  \1-098\

                                                                              .EXTRN  FOR$$ERR_OPECLO
                                                                              .EXTRN  FOR$$SIGNAL_STO
                                                                              .EXTRN  FOR$$SIG_NO_LUB
                                                                              .EXTRN  FOR$$CB_PUSH, FOR$$CB_POP
                                                                              .EXTRN  FOR$$GET_VM, FOR$$FREE_VM
                                                                              .EXTRN  FOR$$SIG_FATINT
                                                                              .EXTRN  FOR$$DECL_EXITH
                                                                              .EXTRN  FOR$$L_XIT_LOCK

                                                                              .PSECT  _FOR$CODE,NOWRT,  SHR,  PIC,2

                                                          003C 00000          .ENTRY  FOR$$OPEN_DEFLT, Save R2,R3,R4,R5              ; 0208
                                  5E       94      AE   9E 00002              MOVAB   -108(SP), SP
   006C   8F             00       6E               00   2C 00006              MOVC5   #0, (SP), #0, #108, OPEN                      ; 0272
                                                   6E      0000D
                         10       AE       04      AC   D0 0000E              MOVL    ACCESS_VAL, OPEN+16                           ; 0278
                         3C       AE       08      AC   D0 00013              MOVL    TYPE_VAL, OPEN+60                             ; 0279
                         14       AE       0C      AC   D0 00018              MOVL    FORM_VAL, OPEN+20                             ; 0280
                                                   5E   DD 0001D              PUSHL   SP                                            ; 0287
                      0000V       CF               01   FB 0001F              CALLS   #1, FOR$$OPEN_PROC
                                                        04 00024              RET                                                   ; 0289

; Routine Size:  37 bytes,    Routine Base:  _FOR$CODE + 0000

;  227          0290  1
```

```
1 GLOBAL ROUTINE FOR$$OPEN_PROC (                          ! Do an OPEN
1         OPEN_ADR)                                        ! Address of OPEN parameter vector
1     : CALL_CCB NOVALUE =
1
1 !++
1 ! ABSTRACT:
1 !
1 !         This routine performs the OPEN for FOR$OPEN and FOR$$OPEN_DEFLT.
1 !         The OPEN parameters have been picked up and placed in a
1 !         longword array. The index is parameter specific. The parameters
1 !         are processed in a logical order which minimizes the
1 !         distance between parameters which depend on each other.
1 !         Each parameter sets an appropriate part of the LUB/ISB/RAB
1 !         control block or the FAB control block.  If the FAB
1 !         has not been allocated, it is allocated.
1 !         Whenever the FAB, RAB, LUB, or ISB
1 !         are allocated they are initially set to 0. Thus, default values
1 !         are often indicated by zero in these structures.
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !         LUB_ADR.mlu.ra              Adr. of LUB/ISB/RAB control block
1 !         OPEN_ADR.mlu.ra             Adr. of OPEN parameter array of
1 !                                     longwords. Index is of form:
1 !                                     OPEN$K_name. A longword value of 0
1 !                                     indiates an omitted keyword.
1 !
1 ! IMPLICIT INPUTS:
1 !
1 !         LUB$V_READ_ONLY             1 if 'READONLY' specified in CALL FDBSET
1 !         LUB$V_DIRECT                1 if specified on previous DEFINEFILE
1 !         LUB$V_OLD_FILE              1 if specified on previous CALL FDBSET
1 !         LUB$V_UNFORMAT              1 if specified on previous DEFINEFILE
1 !         LUB$W_LUN                   FORTRAN logical unit number
1 !         LUB$W_RBUF_SIZE             Size in bytes of record buffer to be allocated
1 !
1 ! IMPLICIT OUTPUTS:
1 !
1 !         LUB$V_READ_ONLY             1 if 'READONLY' present or CALL FDBSET
1 !         LUB$V_DIRECT                1 if ACCESS = 'DIRECT' or DEFINEFILE
1 !         LUB$V_OLD_FILE              1 if TYPE = 'OLD' or CALL FDBSET 'OLD'
1 !         LUB$V_SCRATCH               1 if TYPE = 'SCRATCH'
1 !         LUB$V_PRINT                 1 if DISPOSE = 'PRINT'
1 !         LUB$V_FIXED                 1 if RECORDTYPE = 'FIXED'
1 !         LUB$V_FORMATTED             1 if FORM = 'FORMATTED' or ommitted
1 !         LUB$V_UNFORMAT              1 if FORM = 'UNFORMATTED'
1 !                                     or DEFINEFILE
1 !         LUB$A_ASSOC_VAR             adr. of n if ASSOCIATEVARIABLE = n is present
1 !                                     in OPEN or DEFINEFILE
1 !         LUB$V_ASS_VAR_L             1 if n is longword
1 !         LUB$W_IFI                   RMS internal file id. Needed in case
1 !                                     FORTRAN CLOSE done.
1 !         LUB$W_RBUF_SIZE             Size in bytes of record buffer allocated.
1 !         LUB$L_LOG_RECNO             1
1 !         LUB$W_R_MARGIN              List directed output line width
1 !         LUB$B_ORGAN                 Organization, either LUB$K_ORG_SEQUE
1 !                                     or LUB$K_ORG_RELAT.
```

```
1 !
1 ! COMPLETION STATUS:
1 !
1 !       NONE
1 !
1 ! SIDE EFFECTS:
1 !
1 !       SIGNAL_STOPs the following errors:
1 !       FOR$_FILNOTFOU  (29 = 'FILE NOT FOUND')
1 !       FOR$_OPEFAI     (30 = 'OPEN FAILURE')
1 !       FOR$_INCRECLEN  (37 = 'INCONSISTENT RECORD LENGTH')
1 !       FOR$_INSVIRMEM  (41 = 'INSUFFICIENT VIRTUAL MEMORY)
1 !       FOR$_NO_SUCDEV  (42 = 'NO SUCH DEVICE')
1 !       FOR$_FILNAMSPE  (43 = 'FILE NAME SPECIFICATION ERROR')
1 !       FOR$_RECSPEERR  (44 = 'RECORD SPECIFICATION ERROR')
1 !       FOR$_KEYVALERR  (45 = 'KEYWORD VALUE ERROR IN OPEN STATEMENT')
1 !       FOR$_INCOPECLO  (46 = 'INCONSISTENT OPEN/CLOSE ARGUMENTS')
1 !       FOR$_INVARGFOR  (47 = 'INVALID ARGUMENT TO FORTRAN I/O LIBRARY')
1 !--
1
2     BEGIN
2
2     EXTERNAL REGISTER
2         CCB : REF $FOR$CCB_DECL;
2
2     MAP
2         OPEN_ADR : REF VECTOR [OPEN$K_KEY_MAX + 1];
2
2     LOCAL
2         V_DEFAULT_SIZE,
2         OPEN_STATUS,                              ! RMS status returned on $OPEN or $CREATE
2         T_DFLT_FILE_NAM : VECTOR [10, BYTE],      ! 10-byte  default filename string
2                                                   ! Form: FORnnn.DAT
2         ORIG_RAT: BYTE,                           ! Original FAB$B_RAT
2         XAB_BLOCK : BLOCK [XAB$C_FHCLEN, BYTE],   ! allocate local FHC XAB BLOCK
2         KEY_XAB : REF BLOCK [OPEN$K_XAB_SIZE, BYTE],     ! ISAM key XAB
2         TEMP_FNS: VECTOR [NAM$C_MAXRSS, BYTE],    ! Temp filespec for ASSIGN
2         RES_OR_EXP_NAME : VECTOR [NAM$C_MAXRSS, BYTE];  ! Storage for resultant or expanded name string
2
2     BIND
2         FAB = CCB: REF $FOR$FAB_CCB_STRUCT,       ! FAB is after RAB in CCB
2         NAM = CCB: REF $FOR$NAM_CCB_STRUCT,       ! NAM is after FAB in CCB
2         A_SYS$INPUT = UPLIT BYTE('SYS$INPUT:'),
2         A_SYS$OUTPUT = UPLIT BYTE('SYS$OUTPUT:');
2
2     BUILTIN
2         TESTBITSC;
2
2     LITERAL
2         L_SYS$INPUT = %CHARCOUNT ('SYS$INPUT:'),
2         L_SYS$OUTPUT = %CHARCOUNT ('SYS$OUTPUT:');
2
2     !+
2     ! See if ASSIGN or FDBSET has already allocated us a FAB.  If so,
2     ! copy it to our local FAB and deallocate it.  Copy the filename too
2     ! if it's there.
2     !-
```

```
    IF .CCB [LUB$A_FAB] NEQA 0
    THEN
        BEGIN
        LOCAL
            HEAP_FAB: REF BLOCK [, BYTE];
        HEAP_FAB = .CCB [LUB$A_FAB];
        CH$MOVE (.HEAP_FAB [FAB$B_BLN], .HEAP_FAB, FAB [0,0,0,0]);
        FOR$$FREE_VM (.HEAP_FAB [FAB$B_BLN], .HEAP_FAB);
        CCB [LUB$A_FAB] = 0;
        IF .FAB [FAB$B_FNS] NEQU 0
        THEN
            BEGIN
            CH$MOVE (.FAB [FAB$B_FNS], .FAB [FAB$L_FNA], TEMP_FNS);
            FOR$$FREE_VM (.FAB [FAB$B_FNS], .FAB [FAB$L_FNA]);
            FAB [FAB$L_FNA] = TEMP_FNS;
            END;
        END;

    !+
    ! Initialize NAM and FHC XAB_BLOCK.
    !-

    FAB [FAB$L_NAM] = NAM [0,0,0,0];
    NAM [NAM$L_RSA] = NAM [NAM$L_ESA] = RES_OR_EXP_NAME;
    NAM [NAM$B_RSS] = NAM [NAM$B_ESS] = NAM$C_MAXRSS;
    $XABFHC_INIT (XAB = XAB_BLOCK);
    FAB [FAB$L_XAB] = XAB_BLOCK;
    KEY_XAB = XAB_BLOCK;          ! First XAB in chain

    !+
    ! Set deferred write bit in the FAB for speed improvement in
    ! relative files.
    !-

    FAB [FAB$V_DFW] = 1;
```

```
2 !
2
2       !+
2       ! NAME
2       ! Setup RMS default filename string (FAB$L_DNA, FAB$B_DNS) and
2       ! file name string (FAB$L_FNA) depending on the type of statement
2       ! that caused the LUN to be opened.
2       !
2       !   statement       file name string         default file name string
2       !
2       !   READ            FOR$READ:                FORREAD.DAT
2       !   ACCEPT          FOR$ACCEPT:              FORACCEPT.DAT
2       !   TYPE            FOR$TYPE:                FORTYPE.DAT
2       !   PRINT           FOR$PRINT:               FORPRINT.DAT
2       !   other           FORnnn:                  FORnnn.DAT
2       !
2       ! Get the logical unit number from LUB$W_LUN instead of
2       ! OPEN[OPEN$K_UNIT] since default open doesn't set up UNIT.
2       ! LUN has been checked for being in legal range by CB_PUSH.
2       ! Set the string length and address in the FAB.
2       !-
2
3       BEGIN
3
3       LOCAL
3           A_DEF_LOGNAM,                           ! Address of default logical name
3           L_DEF_LOGNAM;                           ! Length of default logical name
3
3       A_DEF_LOGNAM = 0;                           ! No default yet
3
3       CASE .CCB [LUB$W_LUN] FROM LUB$K_DLUN_MIN TO LUB$K_DLUN_MAX OF
3           SET
3
3           [LUB$K_LUN_READ] :                      ! READ statement (therefore default open)
4               BEGIN
4               FAB [FAB$B_DNS] = %CHARCOUNT ('FORREAD.DAT');
4               FAB [FAB$L_DNA] = UPLIT BYTE('FORREAD.DAT');
4               FAB [FAB$B_FNS] = %CHARCOUNT ('FOR$READ:');
4               FAB [FAB$L_FNA] = UPLIT BYTE('FOR$READ:');
4               A_DEF_LOGNAM = A_SYS$INPUT;
4               L_DEF_LOGNAM = L_SYS$INPUT;
3               END;
3
3           [LUB$K_LUN_ACCE] :                      ! ACCEPT statement (therefore default open)
4               BEGIN
4               FAB [FAB$B_DNS] = %CHARCOUNT ('FORACCEPT.DAT');
4               FAB [FAB$L_DNA] = UPLIT BYTE('FORACCEPT.DAT');
4               FAB [FAB$B_FNS] = %CHARCOUNT ('FOR$ACCEPT:');
4               FAB [FAB$L_FNA] = UPLIT BYTE('FOR$ACCEPT:');
4               A_DEF_LOGNAM = A_SYS$INPUT;
4               L_DEF_LOGNAM = L_SYS$INPUT;
3               END;
3
3           [LUB$K_LUN_TYPE] :                      ! TYPE statement (therefore default open)
4               BEGIN
4               FAB [FAB$B_DNS] = %CHARCOUNT ('FORTYPE.DAT');
4               FAB [FAB$L_DNA] = UPLIT BYTE('FORTYPE.DAT');
```

```
4              FAB [FAB$B_FNS] = %CHARCOUNT ('FOR$TYPE:');
4              FAB [FAB$L_FNA] = UPLIT BYTE('FOR$TYPE:');
4              A_DEF_LOGNAM = A_SYS$OUTPUT;
4              L_DEF_LOGNAM = L_SYS$OUTPUT;
3              END;
3
3          [LUB$K_LUN_PRIN] :                         ! PRINT statement (therefore default open)
4              BEGIN
4              FAB [FAB$B_DNS] = %CHARCOUNT ('FORPRINT.DAT');
4              FAB [FAB$L_DNA] = UPLIT BYTE('FORPRINT.DAT');
4              FAB [FAB$B_FNS] = %CHARCOUNT ('FOR$PRINT:');
4              FAB [FAB$L_FNA] = UPLIT BYTE('FOR$PRINT:');
4              A_DEF_LOGNAM = A_SYS$OUTPUT;
4              L_DEF_LOGNAM = L_SYS$OUTPUT;
3              END;
3
3          [OUTRANGE] :                               ! Some other statement (OPEN or default OPEN)
4              BEGIN
4              IF .OPEN_ADR [OPEN$K_NAME] EQLA 0 OR
4                 .OPEN_ADR [OPEN$K_DEFAULTF] EQLA 0
4              THEN
5                  BEGIN
5                  T_DFLT_FILE_NAM [0] = %C'F';
5                  T_DFLT_FILE_NAM [1] = %C'O';
5                  T_DFLT_FILE_NAM [2] = %C'R';
5                  T_DFLT_FILE_NAM [3] = ((.CCB [LUB$W_LUN]/100) MOD 10) + %C'0';
5                  T_DFLT_FILE_NAM [4] = ((.CCB [LUB$W_LUN]/10) MOD 10) + %C'0';
5                  T_DFLT_FILE_NAM [5] = ((.CCB [LUB$W_LUN]) MOD 10) + %C'0';
5                  T_DFLT_FILE_NAM [6] = %C'.';
5                  T_DFLT_FILE_NAM [7] = %C'D';
5                  T_DFLT_FILE_NAM [8] = %C'A';
5                  T_DFLT_FILE_NAM [9] = %C'T';
4                  END;
4
4              !+
4              ! DEFAULTFILE
4              ! Set up default file name string to be used in RMS $OPEN
4              !-
4
4              IF .OPEN_ADR [OPEN$K_DEFAULTF] NEQA 0
4              THEN
5                  BEGIN
5                  LOCAL
5                      NAM_DSC : REF BLOCK [8, BYTE];
5                  NAM_DSC = .OPEN_ADR [OPEN$K_DEFAULTF];
5                  IF .NAM_DSC [DSC$W_LENGTH] GTRU 255 THEN $FOR$$SIGNAL_STO (FOR$K_FILNAMSPE);
5                  FAB [FAB$B_DNS] = .NAM_DSC [DSC$W_LENGTH];
5                  FAB [FAB$L_DNA] = .NAM_DSC [DSC$A_POINTER];
5                  END
4              ELSE
4
4              !+
4              ! Default file name not specified in OPEN or this is default OPEN.
4              !-
4
5                  BEGIN
5                  FAB [FAB$B_DNS] = %CHARCOUNT ('FORnnn.DAT');
```

```
                  FAB [FAB$L_DNA] = T_DFLT_FILE_NAM;
                  END;

              !+
              ! FILE
              ! Setup file name string to be used in RMS $OPEN
              !-

              IF .OPEN_ADR [OPEN$K_NAME] NEQA 0
              THEN
                  BEGIN

                  !+
                  ! file name specified in OPEN
                  ! Set length and address in FAB
                  !-

                  LOCAL
                      NAM_DSC : REF BLOCK [8, BYTE];        ! File name descriptor

                  NAM_DSC = .OPEN_ADR [OPEN$K_NAME];        ! Get descriptor

                  IF .NAM_DSC [DSC$W_LENGTH] GTRU 255 THEN $FOR$$SIGNAL_STO (FOR$K_FILNAMSPE);

                  FAB [FAB$B_FNS] = .NAM_DSC [DSC$W_LENGTH];
                  FAB [FAB$L_FNA] = .NAM_DSC [DSC$A_POINTER];
                  END
              ELSE

              !+
              ! File name not specified in OPEN or this is default OPEN.
              ! If name not already setup (CALL ASSIGN), use all but last 4 characters of default filename str
              ! i.e., all characters but .DAT
              ! Thus filename string is a string with no punctuation so it can be a logical name
              !-

                  IF .FAB [FAB$L_FNA] EQLA 0
                  THEN
                      BEGIN
                      FAB [FAB$B_FNS] = %CHARCOUNT ('FORnnn');
                      FAB [FAB$L_FNA] = T_DFLT_FILE_NAM;

                      !+
                      ! If this is unit 5 or 6, set up default logical
                      ! name to use if translation of FOR005 or FOR006
                      ! fails.
                      !-

                      IF .CCB [LUB$W_LUN] EQL 5
                      THEN
                          BEGIN
                          A_DEF_LOGNAM = A_SYS$INPUT;
                          L_DEF_LOGNAM = L_SYS$INPUT;
                          END
                      ELSE
```

```
                                IF .CCB [LUB$W_LUN] EQL 6
                                THEN
                                    BEGIN
                                    A_DEF_LOGNAM = A_SYS$OUTPUT;
                                    L_DEF_LOGNAM = L_SYS$OUTPUT;
                                    END;

                            END;

                    END;                                    ! End OUTRANGE expression
                TES;

            !+
            ! If we have an implicit logical name assignment possible
            ! (unit<0 or unit=5 or unit=6) then attempt translation of
            ! the logical name.  If it fails, then substitute the default
            ! logical name SYS$INPUT: or SYS$OUTPUT: appropriately.
            !-

            IF .A_DEF_LOGNAM NEQ 0
            THEN
                BEGIN

                LOCAL
                    LOGNAM_DSC : DSC$DESCRIPTOR,            ! Logical name descriptor
                    RESULT_DSC : DSC$DESCRIPTOR;            ! Translation result descriptor

                LOGNAM_DSC [DSC$B_CLASS] = DSC$K_CLASS_S;
                LOGNAM_DSC [DSC$B_DTYPE] = DSC$K_DTYPE_T;
                RESULT_DSC [DSC$B_CLASS] = DSC$K_CLASS_S;
                RESULT_DSC [DSC$B_DTYPE] = DSC$K_DTYPE_T;
                RESULT_DSC [DSC$W_LENGTH] = NAM$C_MAXRSS;           ! Scratch string
                RESULT_DSC [DSC$A_POINTER] = RES_OR_EXP_NAME;
                LOGNAM_DSC [DSC$A_POINTER] = .FAB [FAB$L_FNA];
                LOGNAM_DSC [DSC$W_LENGTH] = .FAB [FAB$B_FNS];

                IF .CCB [LUB$W_LUN] LSS 0
                THEN

                !+
                ! Don't translate trailing colon.
                !-

                    LOGNAM_DSC [DSC$W_LENGTH] = .LOGNAM_DSC [DSC$W_LENGTH] - 1;

                !+
                ! Attempt to translate the logical name, putting the result in
                ! RES_OR_EXP_NAME.  We don't care what it translated to, just
                ! the fact that it does translate.  If it does not, then substitute
                ! the default logical name for the file name.
                !-

                IF $TRNLOG (LOGNAM = LOGNAM_DSC, RSLBUF = RESULT_DSC) EQLU SS$_NOTRAN
                THEN
                    BEGIN
                    FAB [FAB$L_FNA] = .A_DEF_LOGNAM;
                    FAB [FAB$B_FNS] = .L_DEF_LOGNAM;
```

```
;  4              END;
;  4
;  3          END;
;  3
;  2      END;
;  2
;  2      !+
;  2      ! Set the filename in the LUB in case an error occurs.
;  2      !-
;  2
;  2      CCB [LUB$A_RSN] = .FAB [FAB$L_FNA];
;  2      CCB [LUB$B_RSL] = .FAB [FAB$B_FNS];
;  2 !<BLF/PAGE>
```

```
2      !+
2      ! Do a $PARSE on the file to see if the file is a network file.  If
2      ! so, we will set FAB$V_SQO and not enable RFA cacheing.  Otherwise,
2      ! we'll leave SQO clear so that RFA cacheing can be allowed.
2      !-
2
2      FAB [FAB$L_NAM] = 0;
3      IF $PARSE (FAB = FAB [0,0,0,0])
2      THEN
3          BEGIN
3          BIND
3              FAB_DEV = FAB [FAB$L_DEV]: BLOCK [4, BYTE];
3          IF .FAB_DEV [DEV$V_NET]
3          THEN
3              FAB [FAB$V_SQO] = 1;
2          END;
2      FAB [FAB$L_STS] = 0;          ! Hide error, if any
2      FAB [FAB$L_NAM] = NAM [0,0,0,0];
```

```
2       !+
2       ! READONLY
2       ! Set functions which may be done subsequently (FAB$B_FAC).
2       ! If not READONLY, permit GET, PUT, TRUNCATE (via TPT), UPDATE and DELETE.
2       ! If READONLY, set LUB$V_READ_ONLY bit and use RMS default functions
2       ! which can be done subsequently, namely just GETs.
2       !-
2
2       IF .OPEN_ADR [OPEN$K_READONLY]
2       THEN
3           BEGIN
3           CCB [LUB$V_READ_ONLY] = 1;
3           END
2       ELSE
2
3           IF (.FAB [FAB$B_FAC] EQLU 0)
2           THEN
2               FAB [FAB$B_FAC] = FAB$M_GET + FAB$M_PUT + FAB$M_TRN + FAB$M_DEL + FAB$M_UPD;
2
2       !+
2       ! ACCESS
2       !-
2
2       !+
2       ! If LUB$L_LOG_RECNO is zero, then this is not a default open of
2       ! a direct access file, so set the record number to 1.  Otherwise,
2       ! leave it alone because it has already been set by FOR$$IO_BEG.
2       !-
2       IF .CCB [LUB$L_LOG_RECNO] EQL 0
2       THEN
2           CCB [LUB$L_LOG_RECNO] = 1;
2
2       FAB [FAB$V_NEF] = 1;                              ! inhibit EOF positioning on MT
2
2       CASE .OPEN_ADR [OPEN$K_ACCESS] FROM 0 TO OPEN$K_ACC_KEY OF
2           SET
2
2           [OPEN$K_ACC_DIR] :                            ! ACCESS = 'DIRECT'
3               BEGIN
3               CCB [LUB$V_DIRECT] = 1;
3               FAB [FAB$V_SQO] = 0;                      ! May have been set earlier
3               CCB [RAB$B_RAC] = RAB$C_KEY;
3               CCB [RAB$L_KBF] = CCB [LUB$L_LOG_RECNO];
3               CCB [RAB$B_KSZ] = 0;
3               CCB [RAB$V_UIF] = 1;                      ! Update on $PUT
2               END;
2
2           [0, OPEN$K_ACC_SEQ] :                         ! omitted or ACCESS = 'SEQUENTIAL'
3               BEGIN
3               CCB [LUB$V_SEQUENTIA] = 1;
3               CCB [RAB$B_RAC] = RAB$C_SEQ;
2               END;
2
2           [OPEN$K_ACC_APP] :                            ! ACCESS = 'APPEND'
3               BEGIN
3               IF .CCB [LUB$V_READ_ONLY]
3               THEN
```

```
3                 $FOR$$SIGNAL_STO (FOR$K_INCOPECLO);
3             CCB [RAB$V_EOF] = 1;
3             CCB [LUB$V_APPEND] = 1;
3             FAB [FAB$V_NEF] = 0;                    ! don't inhibit EOF positioning on MT
3             CCB [RAB$B_RAC] = RAB$C_SEQ;
2             END;
2
2         [OPEN$K_ACC_KEY] :                          ! ACCESS = 'KEYED'
3             BEGIN
3             FAB [FAB$V_SQO] = 0;                    ! May have been set earlier
3             CCB [RAB$B_RAC] = RAB$C_KEY;
3             CCB [RAB$B_KRF] = 0;
3             CCB [LUB$V_KEYED] = 1;                  ! So we know later
2             END;
2
2         [OUTRANGE] :
2             $FOR$$SIGNAL_STO (FOR$K_INVARGFOR);
2         TES;
2
2 !<BLF/PAGE>
```

```
2 !
2
2       !+
2       ! TYPE
2       !-
2
2       CASE .OPEN_ADR [OPEN$K_TYPE] FROM 0 TO OPEN$K_TYP_UNK OF
2           SET
2
2           [OPEN$K_TYP_OLD] :                          ! TYPE = 'OLD'
2               CCB [LUB$V_OLD_FILE] = 1;
2
2           [0, OPEN$K_TYP_NEW] :                       ! omitted or TYPE = 'NEW'
3               BEGIN
3               IF NOT .CCB [LUB$V_OLD_FILE]            ! Could have been set by FDBSET
3               THEN
3                   IF .CCB [LUB$V_READ_ONLY] OR
3                      .CCB [LUB$V_APPEND]
3                   THEN
3                       $FOR$$SIGNAL_STO (FOR$K_INCOPECLO);
2               END;
2
2           [OPEN$K_TYP_SCR] :                          ! TYPE = 'SCRATCH'
3               BEGIN
3               CCB [LUB$V_SCRATCH] = 1;
3               FAB [FAB$V_TMD] = 1;
3               IF .CCB [LUB$V_READ_ONLY] OR
3                  .CCB [LUB$V_APPEND]
3               THEN
3                   $FOR$$SIGNAL_STO (FOR$K_INCOPECLO);
2               END;
2
2           [OPEN$K_TYP_UNK] :                          ! TYPE = 'UNKNOWN'
3               BEGIN
3               FAB [FAB$V_CIF] = 1;
3               IF .CCB [LUB$V_READ_ONLY]
3               THEN
3                   $FOR$$SIGNAL_STO (FOR$K_INCOPECLO);
2               END;
2
2           [OUTRANGE] :
2               $FOR$$SIGNAL_STO (FOR$K_INVARGFOR);
2           TES;
2
2 !<BLF/PAGE>
```

```
2 !
2
2       !+
2       ! DISPOSE
2       !    Set bits in LUB to indicate DISPOSE parameters.  Do not allow
2       !    deletion of READONLY or SCRATCH files, printing or submitting of
2       !    SCRATCH files, or saving of SCRATCH files.
2       !-
2
2       SELECT .OPEN_ADR [OPEN$K_DISPOSE] OF
2           SET
2
2           [0] :
2           ;                                               ! ommitted, do nothing
2
2           [OPEN$K_DIS_SAV] :                              ! DISPOSE = 'SAVE'
2
2               IF .CCB [LUB$V_SCRATCH] THEN $FOR$$SIGNAL_STO (FOR$K_INCOPECLO);
2
2           [OPEN$K_DIS_DEL, OPEN$K_DIS_PRDE, OPEN$K_DIS_SUDE] :
2               ! DISPOSE = 'DELETE', 'PRINT/DELETE', 'SUBMIT/DELETE'
3               BEGIN
3               IF .CCB [LUB$V_READ_ONLY]
3               THEN
3                   $FOR$$SIGNAL_STO (FOR$K_INCOPECLO);
3               CCB [LUB$V_DELETE] = 1;
2               END;
2
2           [OPEN$K_DIS_PRI, OPEN$K_DIS_PRDE] :
2               ! DISPOSE = 'PRINT', 'PRINT/DELETE'
3               BEGIN
3
3               IF .CCB [LUB$V_SCRATCH] THEN $FOR$$SIGNAL_STO (FOR$K_INCOPECLO);
3
3               CCB [LUB$V_PRINT] = 1;
2               END;
2
2           [OPEN$K_DIS_SUB, OPEN$K_DIS_SUDE] :
2               ! DISPOSE = 'SUBMIT', 'SUBMIT/DELETE'
3               BEGIN
3
3               IF .CCB [LUB$V_SCRATCH]
3               THEN
4                   $FOR$$SIGNAL_STO (FOR$K_INCOPECLO)
3               ELSE
3                   CCB [LUB$V_SUBMIT] = 1;
3
2               END;
2
2           [OTHERWISE] :
2               $FOR$$SIGNAL_STO (FOR$K_INVARGFOR);
2           TES;
2
2 !<BLF/PAGE>
```

```
!

        !+
        ! FORM
        !-

        CASE .OPEN_ADR [OPEN$K_FORM] FROM OPEN$K_FOR_UNS TO OPEN$K_FOR_UNF OF
            SET

            [OPEN$K_FOR_UNS] :
            ;                                           ! unspecified, used by default OPEN only

            [0] :                                       ! omitted

                IF .CCB [LUB$V_DIRECT] OR .CCB [LUB$V_KEYED]
                THEN
                    CCB [LUB$V_UNFORMAT] = 1
                ELSE
                    CCB [LUB$V_FORMATTED] = 1;

            [OPEN$K_FOR_FOR] :                          ! FORM = 'FORMATTED''
                CCB [LUB$V_FORMATTED] = 1;

            [OPEN$K_FOR_UNF] :                          ! FORM = 'UNFORMATTED'
                CCB [LUB$V_UNFORMAT] = 1;

            [OUTRANGE] :
                $FOR$$SIGNAL_STO (FOR$K_INVARGFOR);
            TES;

!<BLF/PAGE>
```

```
!

!+
! RECORDTYPE
!-

CASE .OPEN_ADR [OPEN$K_RECORDTY] FROM 0 TO OPEN$K_REC_STMLF OF
    SET

    [0] :                                               ! omitted

        !+
        ! Do nothing right now.  We have insufficient information
        ! to determine the recordtype.  Wait until the organization
        ! has been determined.
        !-

    ;

    [OPEN$K_REC_FIX] :                                  ! RECORDTYPE = 'FIXED'
        BEGIN
        CCB [LUB$V_FIXED] = 1;
        FAB [FAB$B_RFM] = FAB$C_FIX;
        END;

    [OPEN$K_REC_VAR] :                                  ! RECORDTYPE = 'VARIABLE'
        BEGIN
        FAB [FAB$B_RFM] = FAB$C_VAR;
        END;

    [OPEN$K_REC_SEGM] :                                 ! RECORDTYPE = 'SEGMENTED'
        BEGIN

        IF .CCB [LUB$V_DIRECT] OR .CCB [LUB$V_KEYED] OR .CCB [LUB$V_FORMATTED]
        THEN
            $FOR$$SIGNAL_STO (FOR$K_INCOPECLO);

        FAB [FAB$B_RFM] = FAB$C_VAR;
        CCB [LUB$V_SEGMENTED] = 1;
        END;

    [OPEN$K_REC_STM] :                                  ! RECORDTYPE = 'STREAM'
        BEGIN
        FAB [FAB$B_RFM] = FAB$C_STM;
        END;

    [OPEN$K_REC_STMCR] :                                ! RECORDTYPE = 'STREAM_CR'
        BEGIN
        FAB [FAB$B_RFM] = FAB$C_STMCR;
        END;

    [OPEN$K_REC_STMLF] :                                ! RECORDTYPE = 'STREAM_LF'
        BEGIN
        FAB [FAB$B_RFM] = FAB$C_STMLF;
        END;

    [OUTRANGE] :
```

```
;  909          0964  2                  $FOR$$SIGNAL_STO (FOR$K_INVARGFOR);
;  910          0965  2          TES;
;  911          0966  2
;  912          0967  2 !<BLF/PAGE>
```

```
2 !
2
2       !+
2       ! CARRIAGECONTROL
2       !-
2
2       CASE .OPEN_ADR [OPEN$K_CARRIAGE] FROM 0 TO OPEN$K_CAR_NON OF
2           SET
2
2           [0] :                                       ! omitted
2
2               IF .CCB [LUB$V_FORMATTED] THEN FAB [FAB$V_FTN] = 1;
2
2           [OPEN$K_CAR_FOR] :                          ! CARRIAGECONTROL = 'FORTRAN'
2               FAB [FAB$V_FTN] = 1;
2
2           [OPEN$K_CAR_LIS] :                          ! CARRIAGECONTROL = 'LIST'
2               FAB [FAB$V_CR] = 1;
2
2           [OPEN$K_CAR_NON] :
2           ;                                           ! CARRIAGECONTROL = 'NONE', do nothing
2
2           [OUTRANGE] :
2               $FOR$$SIGNAL_STO (FOR$K_INVARGFOR);
2           TES;
2
2           !+
2           ! Store FAB$B_RAT so we can "restore" it if we find we've
2           ! opened a process-permanent file.
2           !-
2
2           ORIG_RAT = .FAB [FAB$B_RAT];
2
2 !<BLF/PAGE>
```

```
!

        !+
        ! ORGANIZATION
        !-

        CCB [LUB$V_NOTSEQORG] = 1;                      ! Assume not sequential organization

        CASE .OPEN_ADR [OPEN$K_ORGANIZA] FROM 0 TO OPEN$K_ORG_IDX OF
            SET

            [0, OPEN$K_ORG_SEQ] :                       ! omitted or ORGANIZATION = ;SEQUENTIAL'
                BEGIN

                IF .CCB [LUB$V_DIRECT] AND .FAB [FAB$B_RFM] EQLU FAB$C_VAR THEN $FOR$$SIGNAL_STO (FOR$K_INCOPECL

                ;

                IF .CCB [LUB$V_KEYED] AND .OPEN_ADR [OPEN$K_ORGANIZA] NEQ 0
                THEN
                    $FOR$$SIGNAL_STO (FOR$K_INCOPECLO);

                FAB [FAB$B_ORG] = FAB$C_SEQ;
                CCB [LUB$V_NOTSEQORG] = 0;              ! So ENDFILE will know its sequential
                END;

            [OPEN$K_ORG_REL] :                          ! ORGANIZATION = 'RELATIVE'
                BEGIN

                IF .CCB [LUB$V_SEGMENTED] OR .CCB [LUB$V_KEYED] THEN $FOR$$SIGNAL_STO (FOR$K_INCOPECLO);

                FAB [FAB$B_ORG] = FAB$C_REL;
                END;

            [OPEN$K_ORG_IDX] :                          ! ORGANIZATION = 'INDEXED'
                BEGIN

                IF .CCB [LUB$V_DIRECT] OR .CCB [LUB$V_APPEND] OR .CCB [LUB$V_SEGMENTED]
                THEN
                    $FOR$$SIGNAL_STO (FOR$K_INCOPECLO);

                FAB [FAB$B_ORG] = FAB$C_IDX;
                END;

            [OUTRANGE] :
                $FOR$$SIGNAL_STO (FOR$K_INVARGFOR);
            TES;

        !+
        ! Verify that user didn't ask for a non-sequential stream file.
        !-

        IF .CCB [LUB$V_NOTSEQORG] AND
            ONE_OF (.FAB [FAB$B_RFM], FAB$C_STM, FAB$C_STMCR, FAB$C_STMLF)
        THEN
            $FOR$$SIGNAL_STO (FOR$K_INCOPECLO);
```

```
2       !+
2       ! RECORDTYPE continued
2       ! We now have enough information to determine the initial recordtype
2       ! if it was omitted.
2       !-
2
2       IF .OPEN_ADR [OPEN$K_RECORDTY] EQL 0
2       THEN
2
2           IF .FAB [FAB$B_ORG] EQL FAB$C_REL OR .FAB [FAB$B_ORG] EQL FAB$C_IDX OR .CCB [LUB$V_DIRECT] OR .CCB [
2                   LUB$V_KEYED]
2           THEN
3               BEGIN
3               FAB [FAB$B_RFM] = FAB$C_FIX;
3               CCB [LUB$V_FIXED] = 1;
3               END
2           ELSE
3               BEGIN
3               FAB [FAB$B_RFM] = FAB$C_VAR;
3
3               IF .CCB [LUB$V_UNFORMAT] THEN CCB [LUB$V_SEGMENTED] = 1;
3
2               END;
2
2       !+
2       ! SHARED
2       ! If SHARED, indicate user provided record interlock (UPI) (for SEQUENTIAL ORG only)
2       ! If not SHARED, RMS defaults is read, sharing only if READONLY, else no sharing.
2       !-
2
2       IF .OPEN_ADR [OPEN$K_SHARED]
2       THEN
3           BEGIN
3           FAB [FAB$B_SHR] = FAB$M_SHRGET + FAB$M_SHRPUT + FAB$M_SHRUPD + FAB$M_SHRDEL;
3
3           IF NOT .CCB [LUB$V_NOTSEQORG]           ! Sequential only, set UPI
3           THEN
3               FAB [FAB$V_UPI] = 1;
3
2           END;
2
2 !<BLF/PAGE>
```

```
2 !
2
2      !+
2      ! KEY
2      !-
2
2      IF .OPEN_ADR [OPEN$K_KEY] NEQU 0
2      THEN
3          BEGIN
3
3          LOCAL
3              KEY_DEFN : REF BLOCK [12, BYTE],     ! Key definition
3              KEY_NUM,                              ! Number of current key
3              KEY_COUNT,                            ! Total number of keys defined
3              XAB_ADDR;                             ! Address of newly allocated KEY XAB
3
3          IF .FAB [FAB$B_ORG] NEQU FAB$C_IDX THEN $FOR$$SIGNAL_STO (FOR$K_INCOPECLO);
3
3          KEY_DEFN = .OPEN_ADR [OPEN$K_KEY];
3          KEY_COUNT = .KEY_DEFN [OPEN$W_INFO];
3          KEY_DEFN = .KEY_DEFN + %UPVAL;
3
3          IF .KEY_COUNT MOD 3 NEQ 0 THEN $FOR$$SIGNAL_STO (FOR$K_INVARGFOR);
3
3          KEY_COUNT = .KEY_COUNT/3;
3
3          !+
3          ! Loop through key definitions, and set up the key XABs.
3          !-
3
3          INCR KEY_NUM FROM 0 TO .KEY_COUNT - 1 DO
4              BEGIN
4              XAB_ADDR = FOR$$GET_VM (OPEN$K_XAB_SIZE);
4              KEY_XAB [XAB$L_NXT] = .XAB_ADDR;
4              KEY_XAB = .XAB_ADDR;
4
4              !+
4              ! Fill in KEY XAB fields
4              !-
4
4              CH$FILL (0, OPEN$K_XAB_SIZE, .KEY_XAB);
4              KEY_XAB [XAB$B_COD] = XAB$C_KEY;
4              KEY_XAB [XAB$B_BLN] = XAB$C_KEYLEN;
4
4              !+
4              ! Calculate key position and width
4              !-
4
4              IF .KEY_DEFN [OPEN$L_KEY_LO] LEQ 0 OR
4                  .KEY_DEFN [OPEN$L_KEY_LO] GTR 32767 OR
4                  .KEY_DEFN [OPEN$L_KEY_HI] GTR 32767 OR
4                  .KEY_DEFN [OPEN$L_KEY_HI] LSS .KEY_DEFN [OPEN$L_KEY_LO]
4              THEN
4                  $FOR$$SIGNAL_STO (FOR$K_INVKEYSPE);
4
4              KEY_XAB [XAB$W_POS0] = .KEY_DEFN [OPEN$L_KEY_LO] - 1;
4              KEY_XAB [XAB$B_SIZ0] =
                        BEGIN

                        LOCAL
                            SIZE;

                        SIZE = .KEY_DEFN [OPEN$L_KEY_HI] - .KEY_DEFN [OPEN$L_KEY_LO] + 1;

                        IF .SIZE GTR 255 THEN $FOR$$SIGNAL_STO (FOR$K_INVKEYSPE);

                        .SIZE
                        END;
                    KEY_XAB [OPEN$W_POS0] = .KEY_XAB [XAB$W_POS0];
                    KEY_XAB [OPEN$B_SIZ0] = .KEY_XAB [XAB$B_SIZ0];
                    KEY_XAB [XAB$B_DTP] = (SELECTONE .KEY_DEFN [OPEN$B_DTYPE] OF
                        SET
                        [0, DSC$K_DTYPE_T] : XAB$C_STG;
                        [DSC$K_DTYPE_WU] : XAB$C_BN2;
                        [DSC$K_DTYPE_W] : XAB$C_IN2;
                        [DSC$K_DTYPE_LU] : IF .KEY_XAB [XAB$B_SIZ0] EQL 4 THEN XAB$C_BN4 ELSE XAB$C_BN2;
                        [DSC$K_DTYPE_L] : IF .KEY_XAB [XAB$B_SIZ0] EQL 4 THEN XAB$C_IN4 ELSE XAB$C_IN2;
                        [OTHERWISE] :
                            BEGIN
                            $FOR$$SIGNAL_STO (FOR$K_INVARGFOR);
                            END;
                        TES);
                    KEY_XAB [OPEN$B_KTYPE] = .KEY_XAB [XAB$B_DTP];

                    IF .KEY_NUM NEQ 0
                    THEN
                        BEGIN
                        KEY_XAB [XAB$V_CHG] = 1;
                        KEY_XAB [XAB$V_DUP] = 1;
                        END;

                    KEY_XAB [XAB$B_REF] = .KEY_NUM;
                    KEY_DEFN = .KEY_DEFN + (3*%UPVAL);  ! Go to next definition
                    END;

            END;

        !+
        ! BLANK
        !   If user specifies BLANK='NULL' then set LUB$V_NULLBLNK
        !   else leave it alone.
        !-

        CASE .OPEN_ADR [OPEN$K_BLANK] FROM 0 TO OPEN$K_BLK_NUL OF
            SET

            [0, OPEN$K_BLK_ZER] :
            ;                                       ! Do nothing, ZERO is the default

            [OPEN$K_BLK_NUL] :
                CCB [LUB$V_NULLBLNK] = 1;

            [OUTRANGE] :
                $FOR$$SIGNAL_STO (FOR$K_INVARGFOR);
```

```
; 1163          1215  2          TES;
; 1164          1216  2
; 1165          1217  2 !<BLF/PAGE>
```

```
2 !
2
2       !+
2       ! RECORDSIZE
2       !  Set maximum record size (FAB$W_MRS) if fixed, relative, or indexed.
2       !  Set V_DEFAULT_SIZE if omitted.  Set LUB$W_RBUF_SIZE to record size.
2       ! Default is 128 for unformatted fixed length, 2044 for unformatted
2       ! variable length (4 bytes for RMS control info to make total 2048),
2       ! and 133 for formatted (line printer width) or unspecified (ENDFILE
2       ! default OPEN).
2       !-
2
2       V_DEFAULT_SIZE = 0;                             ! assume user specifies
2
2       SELECTONEU .OPEN_ADR [OPEN$K_RECORDSI] OF
2           SET
2
2           [0] :
2
2               !+
2               !   If this is a fixed length or relative file, and
2               !   is not known to exist, RECORDSIZE must be given, else
2               !   error FOR$_INCRECLEN.
2               !-
2
2               IF .CCB [LUB$W_RBUF_SIZE] EQLU 0
2               THEN
3                   BEGIN
3
4                   IF NOT .CCB [LUB$V_OLD_FILE] AND (.CCB [LUB$V_FIXED]
4                     OR .FAB [FAB$B_ORG] EQL FAB$C_REL)
3                   THEN
3                       $FOR$$SIGNAL_STO (FOR$K_INCRECLEN);
3
4                   CCB [LUB$W_RBUF_SIZE] = (
4                       IF .CCB [LUB$V_UNFORMAT]    ! unformatted
4                       THEN
4                           IF .CCB [LUB$V_FIXED]
4                           THEN
4                               128                 !    fixed
4                           ELSE
4                               2044                !    variable
4                       ELSE                        ! formatted or unspecified (ENDFILE default open)
3                           133);
3                   V_DEFAULT_SIZE = 1;             ! user took the default
2                   END;
2
2           [1 TO 32767] :
3               BEGIN
3
3               LOCAL
3                   T;
3
4               T = .OPEN_ADR [OPEN$K_RECORDSI]*(IF .CCB [LUB$V_UNFORMAT] THEN %UPVAL ELSE 1)       !
3               + (IF .CCB [LUB$V_SEGMENTED] THEN 2 ELSE 0);
3
3               IF .T GTRU 32767 THEN $FOR$$SIGNAL_STO (FOR$K_INCRECLEN);
```

```
; 1224          1275  3
; 1225          1276  3                 CCB [LUB$W_RBUF_SIZE] = .T;
; 1226          1277  2                 END;
; 1227          1278  2
; 1228          1279  2             [OTHERWISE] :
; 1229          1280  2                 $FOR$$SIGNAL_STO (FOR$K_INCRECLEN);
; 1230          1281  2             TES;
; 1231          1282  2
; 1232          1283  2         IF .CCB [LUB$V_FIXED]
; 1233          1284  3           OR (.FAB [FAB$B_ORG] EQLU FAB$C_REL)
; 1234          1285  3           OR (.FAB [FAB$B_ORG] EQLU FAB$C_IDX)
; 1235          1286  2         THEN FAB [FAB$W_MRS] = .CCB [LUB$W_RBUF_SIZE];
; 1236          1287  2
; 1237          1288  2 !<BLF/PAGE>
```

```
!

        !+
        ! INITIALSIZE
        ! Only set if specified in explicit OPEN, since may be set by FDBSET on default OPEN.
        !-

        IF .OPEN_ADR [OPEN$K_INITIALS] NEQ 0
        THEN
            BEGIN
            FAB [FAB$L_ALQ] = ABS (.OPEN_ADR [OPEN$K_INITIALS]);
            FAB [FAB$V_CBT] = 1;
            END;

        !+
        ! EXTENDSIZE
        ! Only set if specified explicitly in explicit OPEN, since FDBSET could set on default open.
        !-

        IF .OPEN_ADR [OPEN$K_EXTENDSI] NEQU 0
        THEN

            IF ABS (.OPEN_ADR [OPEN$K_EXTENDSI]) LSSU 1^16
            THEN
                FAB [FAB$W_DEQ] = ABS (.OPEN_ADR [OPEN$K_EXTENDSI])
            ELSE
                $FOR$$SIGNAL_STO (FOR$K_KEYVALERR);

        !+
        ! NOSPANBLOCKS
        !-

        FAB [FAB$V_BLK] = .OPEN_ADR [OPEN$K_NOSPANBL];

        !+
        ! MAXREC
        ! Only set if explicitly passed by OPEN statement, since
        ! DEFINE FILE could have pre-set it if this is default open.
        !-

        IF .OPEN_ADR [OPEN$K_MAXREC] NEQU 0 THEN CCB [LUB$L_REC_MAX] = .OPEN_ADR [OPEN$K_MAXREC];

        FAB [FAB$L_MRN] = .CCB [LUB$L_REC_MAX];
!<BLF/PAGE>
```

```
    2 !
    2
    2       !+
    2       ! BLOCKSIZE
    2       ! Set BLOCKSIZE (used for magtape only), multi-block count (sequential org only)
    2       ! and bucket size (relative/indexed only).
    2       !-
    2
    2       SELECTONEU .OPEN_ADR [OPEN$K_BLOCKSIZ] OF
    2           SET
    2
    2           [0] :
    2           ;                                       ! Use process/system defaults
    2
    2           [1 TO 65535] :
    3               BEGIN
    3               FAB [FAB$W_BLS] = .OPEN_ADR [OPEN$K_BLOCKSIZ];
    3               CCB [RAB$B_MBC] = (.OPEN_ADR [OPEN$K_BLOCKSIZ] + 511)/512;
    3               FAB [FAB$B_BKS] = .CCB [RAB$B_MBC];
    3               IF .FAB [FAB$B_BKS] GTRU 63 ! RMS limit
    3               THEN
    3                   FAB [FAB$B_BKS] = 63;
    2               END;
    2
    2           [OTHERWISE] :
    2               $FOR$$SIGNAL_STO (FOR$K_KEYVALERR);
    2           TES;
    2
    2 !<BLF/PAGE>
```

```
!

        !+
        ! BUFFERCOUNT
        ! Only set if explicitly passed by OPEN statement since FDBSET could
        ! have pre-set it if this is a default open.
        !-

        SELECTONEU .OPEN_ADR [OPEN$K_BUFFERCO] OF
            SET

            [0] :
            ;

            [1 TO 127] :
                CCB [RAB$B_MBF] = .OPEN_ADR [OPEN$K_BUFFERCO];

            [OTHERWISE] :
                $FOR$$SIGNAL_STO (FOR$K_KEYVALERR);
            TES;

        !+
        ! ASSOCIATEVARIABLE
        !-

        IF .OPEN_ADR [OPEN$K_ASSOCIAT] NEQA 0
        THEN
            BEGIN
            CCB [LUB$A_ASSOC_VAR] = .OPEN_ADR [OPEN$K_ASSOCIAT];

            IF .OPEN_ADR [OPEN$K_ASSOC_L] THEN CCB [LUB$V_ASS_VAR_L] = 1

            END;
```

```
2 !
2
2         !+
2         ! USEROPEN
2         !
2         ! If a USEROPEN procedure address was specified then call the procedure
2         ! to do the $OPEN and $CONNECT; it will return an RMS status code as
2         ! procedure value.  Otherwise do the $OPEN and $CONNECT ourselves.
2         ! Set useropen flag, just as a debugging aid in case we get a dump with an SPR.
2         !-
2
2         IF .OPEN_ADR [OPEN$K_USEROPEN] NEQA 0
2         THEN
3             BEGIN
3
3             LOCAL
3                 LOG_UNIT;                               ! Logical unit number
3
3             LOG_UNIT = .CCB [LUB$W_LUN];                ! Get the unit number
3             CCB [LUB$V_USEROPEN] = 1;                   ! so we know the user opened the file!
3             OPEN_STATUS = (.OPEN_ADR [OPEN$K_USEROPEN]) (FAB [0,0,0,0],
3                 .CCB, LOG_UNIT);
3             END
2         ELSE
3             BEGIN                                       ! not USEROPEN
3
3             !+
3             ! If old file is explicitly wanted, do an $OPEN. Otherwise
3             ! (NEW, SCRATCH, UNKNOWN, default = NEW) do a $CREATE.
3             ! UNKNOWN has set RMS FAB$V_CIF to do an OPEN if file
3             ! exists rather than a $CREATE.  If file already existed
3             ! on $CREATE (TYPE='UNKNOWN'), set LUB$V_OLD_FILE
3             ! as flag that file already existed for error checking below.
3             !-
3
4             OPEN_STATUS = (
4                 IF .CCB [LUB$V_OLD_FILE]
4                 THEN
5                     $OPEN (FAB = FAB [0,0,0,0])
4                 ELSE
3                     $CREATE (FAB = FAB [0,0,0,0]));
3
3             !+
3             ! If no error in open/create, do $CONNECT (pointer to FAB already set in RAB).
3             !-
3
3             IF .OPEN_STATUS THEN OPEN_STATUS = $CONNECT (RAB = .CCB);
3
2             END;
2
2 !<BLF/PAGE>
```

```
!

    !+
    ! Zero the XAB pointer in the FAB so we don't accidentally use it later.
    !-

    FAB [FAB$L_XAB] = 0;

    !+
    ! TYPE = 'UNKNOWN' has set RMS FAB$V_CIF to do an open if file exitsts
    ! rather than a create.  If file already existed on $CREATE (TYPE='UNKNOWN'),
    ! set LUB$V_OLD_FILE as flag that file already existed for error checking below.
    !-

    IF .FAB [FAB$V_CIF] AND .FAB [FAB$L_STS] NEQU RMS$_CREATED THEN CCB [LUB$V_OLD_FILE] = 1;

    !+
    ! If CALL ASSIGN allocated space for the filename, deallocate it.
    !-

    IF TESTBITSC (CCB [LUB$V_VIRT_RSN])
    THEN
        FOR$$FREE_VM (.CCB [LUB$B_RSL], .CCB [LUB$A_RSN]);

    !+
    ! If we have an expanded name string (or even better, a resultant name string),
    ! point the LUB to it instead of the user supplied name.  This will be
    ! the file name used for error messages from now on.
    !-

    IF .NAM [NAM$B_RSL] NEQ 0
    THEN
        BEGIN
        CCB [LUB$A_RSN] = .NAM [NAM$L_RSA];
        CCB [LUB$B_RSL] = .NAM [NAM$B_RSL];
        END
    ELSE

        IF .NAM [NAM$B_ESL] NEQ 0
        THEN
            BEGIN
            CCB [LUB$A_RSN] = .NAM [NAM$L_ESA];
            CCB [LUB$B_RSL] = .NAM [NAM$B_ESL];
            END;

!<BLF/PAGE>
```

```
!

    !+
    ! If OPEN or CREATE error, SIGNAL STOP one of:
    ! FOR$_FILNOTFOU (29='FILE NOT FOUND') or
    ! FOR$_OPEFAI (30='OPEN FAILURE')
    ! FOR$_INCRECLEN (37='INCONSISTENT RECORD LENGTH')
    ! FOR$_NO_SUCDEV (42='NO SUCH DEVICE')
    ! FOR$_FILNAMSPE (43='FILE NAME SPECIFICATION ERROR)
    ! FOR$_INVKEYSPE (49='INVALID KEY SPECIFICATION')
    ! Note: OPEN_STATUS can be anything for USEROPEN, so use status in FAB.
    !-

    IF NOT .OPEN_STATUS
    THEN
        $FOR$$SIGNAL_STO (

            (SELECTONEU .FAB [FAB$L_STS] OF
                SET

                [RMS$_FNF] :
                    FOR$K_FILNOTFOU;                ! FILE NOT FOUND

                [RMS$_DEV] :
                    FOR$K_NO_SUCDEV;                ! NO SUCH DEVICE

                [RMS$_FNM, RMS$_NOD, RMS$_TYP, RMS$_VER, RMS$_SYN] :
                    FOR$K_FILNAMSPE;                ! FILE NAME SPECIFICATION ERROR

                [RMS$_POS, RMS$_SIZ, RMS$_NPK] :
                    FOR$K_INVKEYSPE;                ! INVALID KEY SPECIFICATION

                [RMS$_CRE]:

                    !+
                    ! Check for the special case of a mag tape file with
                    ! blocksize less than recordsize (+ 4 if variable).
                    ! If so, signal INCRECLEN, since RMS does not give a
                    ! useful message in this case; otherwise OPEFAI.
                    !-

                    BEGIN
                    LOCAL
                        OLD_STS,        ! Previous FAB$L_STS
                        OLD_STV;        ! Previous STV
                    OLD_STS = .FAB [FAB$L_STS];
                    OLD_STV = .FAB [FAB$L_STV];
                    IF $PARSE (FAB = FAB [0,0,0,0])     ! Get device characteristics

                    THEN
                        BEGIN
                        FAB [FAB$L_STS] = .OLD_STS;
                        FAB [FAB$L_STV] = .OLD_STV;
                        IF .BLOCK [FAB [FAB$L_DEV], DEV$V_SQD; 1, LONG] AND .FAB [FAB$W_BLS] NEQ 0
                                                    ! If mag tape,
                        THEN
                            IF .FAB [FAB$W_BLS] LSSU .CCB [LUB$W_RBUF_SIZE]
; 1506          P 1551  2                                           + (IF NOT .CCB [LUB$V_FIXED] THEN 4 ELSE 0)
; 1507          P 1552  2                                   THEN
; 1508          P 1553  2                                       FOR$K_INCRECLEN ! INCONSISTENT RECORD LENGTH
; 1509          P 1554  2                                   ELSE
; 1510          P 1555  2                                       FOR$K_OPEFAI    ! OPEN FAILURE
; 1511          P 1556  2                               ELSE
; 1512          P 1557  2                                   FOR$K_OPEFAI
; 1513          P 1558  2                               END
; 1514          P 1559  2                           ELSE
; 1515          P 1560  2                               FOR$K_OPEFAI
; 1516          P 1561  2                           END;
; 1517          P 1562  2
; 1518          P 1563  2                       [OTHERWISE]:
; 1519          P 1564  2                           FOR$K_OPEFAI;
; 1520          P 1565  2
; 1521            1566  2                       TES));
; 1522            1567  2
; 1523            1568  2 !<BLF/PAGE>
```

```
!

    !+
    ! If the file we just opened was an existing file, perform a couple of
    ! consistency checks.
    !-

    IF .CCB [LUB$V_OLD_FILE]
    THEN
        BEGIN

        !+
        ! Organization check:
        ! If user program did not specify organization with this OPEN,
        ! use the attributes from the file.  If the user program did specify,
        ! check that it agrees with the file.
        !-

        IF .OPEN_ADR [OPEN$K_ORGANIZA] NEQ 0
        THEN
            BEGIN

            LOCAL
                T;

            T = (CASE .OPEN_ADR [OPEN$K_ORGANIZA] FROM OPEN$K_ORG_SEQ TO OPEN$K_ORG_IDX OF
                SET
                [OPEN$K_ORG_SEQ] : FAB$C_SEQ;
                [OPEN$K_ORG_REL] : FAB$C_REL;
                [OPEN$K_ORG_IDX] : FAB$C_IDX;
                [OUTRANGE] :
                    BEGIN
                    $FOR$$SIGNAL_STO (FOR$K_INVARGFOR);
                    END;
                TES);

            IF .T NEQ .FAB [FAB$B_ORG] THEN $FOR$$SIGNAL_STO (FOR$K_INCFILORG);

            END;

        !+
        ! If ACCESS='KEYED' was specified and the file is not indexed,
        ! signal an error.
        !-

        IF (.CCB [LUB$V_KEYED] AND .FAB [FAB$B_ORG] NEQ FAB$C_IDX) OR (.CCB [LUB$V_DIRECT] AND .FAB [
                FAB$B_ORG] EQL FAB$C_IDX)
        THEN
            $FOR$$SIGNAL_STO (FOR$K_INCFILORG);

!+
! If the file does not have sequential organization, then set LUB bit.
!-

        IF (.FAB [FAB$B_ORG] NEQ FAB$C_SEQ) THEN CCB [LUB$V_NOTSEQORG] = 1;

!<BLF/PAGE>
```

```
!

            !+
            ! Record type check:
            ! If user-program did not specified record-type in this OPEN,
            ! use the file attributes. If user-program did specify
            ! this OPEN, check that it agrees with the file.
            !-

            CASE .OPEN_ADR [OPEN$K_RECORDTY] FROM 0 TO OPEN$K_REC_STMLF OF
                SET

                [0] :                                   ! User did not specify
                    BEGIN
                    CCB [LUB$V_FIXED] = 0;              ! Clear previously set bits
                    CCB [LUB$V_SEGMENTED] = 0;

                    IF .FAB [FAB$B_RFM] EQL FAB$C_FIX
                    THEN
                        CCB [LUB$V_FIXED] = 1           ! Fixed
                    ELSE
                        BEGIN                           ! Variable
                        IF .CCB [LUB$V_DIRECT] AND NOT .CCB [LUB$V_NOTSEQORG]
                        THEN
                            $FOR$$SIGNAL_STO (FOR$K_INCRECTYP);
                        IF NOT .CCB [LUB$V_NOTSEQORG] AND .CCB [LUB$V_UNFORMAT] AND
                            NOT .CCB [LUB$V_DIRECT] AND (.FAB [FAB$B_RFM] EQL FAB$C_VAR)
                        THEN
                            CCB [LUB$V_SEGMENTED] = 1;
                        END;
                    END;

                [OPEN$K_REC_FIX] :

                    IF .FAB [FAB$B_RFM] NEQU FAB$C_FIX THEN $FOR$$SIGNAL_STO (FOR$K_INCRECTYP);

                [OPEN$K_REC_VAR] :

                    IF .FAB [FAB$B_RFM] NEQU FAB$C_VAR AND .FAB [FAB$B_RFM] NEQU FAB$C_VFC
                    THEN
                        $FOR$$SIGNAL_STO (FOR$K_INCRECTYP);

                [OPEN$K_REC_SEGM] :

                    IF (.FAB [FAB$B_RFM] NEQU FAB$C_VAR) OR .CCB [LUB$V_NOTSEQORG]
                    THEN
                        $FOR$$SIGNAL_STO (FOR$K_INCRECTYP);

                [OPEN$K_REC_STM] :

                    IF .FAB [FAB$B_RFM] NEQU FAB$C_STM
                    THEN
                        $FOR$$SIGNAL_STO (FOR$K_INCRECTYP);

                [OPEN$K_REC_STMCR] :

                    IF .FAB [FAB$B_RFM] NEQU FAB$C_STMCR
```

```
                  THEN
                      $FOR$$SIGNAL_STO (FOR$K_INCRECTYP);

              [OPEN$K_REC_STMLF] :

                  IF .FAB [FAB$B_RFM] NEQU FAB$C_STMLF
                  THEN
                      $FOR$$SIGNAL_STO (FOR$K_INCRECTYP);

              [OUTRANGE] :
                  $FOR$$SIGNAL_STO (FOR$K_INVARGFOR);
              TES;

          !+
          ! Set maximum record number from file.
          !-

          IF .CCB [LUB$L_REC_MAX] EQL 0
          THEN
              CCB [LUB$L_REC_MAX] = .FAB [FAB$L_MRN]
          ELSE

              IF .FAB [FAB$L_MRN] NEQ 0 THEN CCB [LUB$L_REC_MAX] = MIN (.CCB [LUB$L_REC_MAX], .FAB [FAB$L_MRN]

          ;
 !<BLF/PAGE>
```

```
3 !
3
3               !+
3               ! Record size check:
3               !       If user specified a record size (with DEFINE FILE or RECORDSIZE
3               !       OPEN keyword, and MRS was required by RMS (fixed or relative),
3               !       or organization indexed and MRS is non-zero, then they must agree.
3               !       The recordsize the OTS will use is then computed in a reasonable
3               !       manner.
3               !-
3
3               !+
3               ! If not a disk or terminal, use the blocksize as the maximum recordsize
3               ! (if not there already).
3               !-
3               IF (NOT .BLOCK [FAB [FAB$L_DEV], DEV$V_RND;4, BYTE]) AND
4                  (NOT .BLOCK [FAB [FAB$L_DEV], DEV$V_TRM;4, BYTE])
3               THEN
3                   IF .FAB [FAB$W_MRS] EQL 0
3                   THEN
3                       FAB [FAB$W_MRS] = .FAB [FAB$W_BLS];
3
4               IF NOT .V_DEFAULT_SIZE AND (.CCB [LUB$V_FIXED]
4                 OR .FAB [FAB$B_ORG] EQL FAB$C_REL)
3               THEN
3
3                   IF .CCB [LUB$W_RBUF_SIZE] NEQU .FAB [FAB$W_MRS] THEN $FOR$$SIGNAL_STO (FOR$K_INCRECLEN);
3
4               IF (.CCB [LUB$V_FIXED]
4                 OR .FAB [FAB$B_ORG] EQL FAB$C_REL)
3               THEN
3                   CCB [LUB$W_RBUF_SIZE] = .FAB [FAB$W_MRS]
3               ELSE
3                   CCB [LUB$W_RBUF_SIZE] = MAXU (.CCB [LUB$W_RBUF_SIZE], .FAB [FAB$W_MRS], .XAB_BLOCK [XAB$W_LRL]);
3
4               IF (.FAB [FAB$B_ORG] EQLU FAB$C_IDX) AND (NOT .CCB [LUB$V_FIXED])
3               THEN
3 !+
3 ! For variable indexed files, determine if the MRS is zero.  If it is, this is an ISAM file
3 ! created prior to FORTRAN V3 and should not be checked for buffer size agreement.
3 ! If no explicit RECL was specified, use the bucketsize to compute the buffersize.
3 !-
3                   IF .FAB [FAB$W_MRS] EQLU 0
3                   THEN
4                       BEGIN
4                       IF .V_DEFAULT_SIZE
4                       THEN
4                           CCB [LUB$W_RBUF_SIZE] = .FAB [FAB$B_BKS] * 512;
4                       END
3                   ELSE
3 !+
3 ! This is a new ISAM file.  Check to be sure that the buffer size requested does
3 ! not exceed the Max Recordsize specified when the file was created.  Set the
3 ! buffer size to the MRS to allow the records to grow.
3 !-
3                       IF NOT .V_DEFAULT_SIZE AND
4                               (.CCB [LUB$W_RBUF_SIZE] GTRU .FAB [FAB$W_MRS])
```

```
                THEN
                    $FOR$$SIGNAL_STO (FOR$K_INCRECLEN)
                ELSE
                    CCB [LUB$W_RBUF_SIZE] = .FAB [FAB$W_MRS];
        !+
        ! Key definition check.  If file is ORGANIZATION='INDEXED' and
        ! user specified a KEY definition, make sure it agrees with
        ! what the file actually has.  Key sizes must match, and key
        ! datatypes must
        ! match.  If not, signal error FOR$_INVKEYSPE.
        ! Make sure that we don't interfere with key XAB's that a
        ! USEROPEN might have defined.
        !-

        IF .FAB [FAB$B_ORG] EQL FAB$C_IDX
        THEN
            BEGIN       ! Indexed file

            LOCAL
                XAB_STATUS,                 ! Status while freeing XABs
                KEY_COUNT;                  ! Count of OPEN defined keys

            BEGIN

            LOCAL
                KEY_DEFN : REF BLOCK [12, BYTE];

            KEY_DEFN = .OPEN_ADR [OPEN$K_KEY];

            IF .KEY_DEFN NEQ 0 THEN KEY_COUNT = .KEY_DEFN [OPEN$W_INFO] ELSE KEY_COUNT = 0;

            END;

            XAB_STATUS=SS$_NORMAL;
            KEY_XAB = .XAB_BLOCK [XAB$L_NXT];

            WHILE .KEY_XAB NEQU 0 AND .KEY_COUNT GTR 0 DO
                BEGIN   ! Go through XABs

                IF (.KEY_XAB [XAB$B_COD] EQL XAB$C_KEY)
                THEN
                    BEGIN

                    IF (.KEY_XAB [XAB$W_POS0] NEQ .KEY_XAB [OPEN$W_POS0]) OR (.KEY_XAB [XAB$B_SIZ0] NEQ
                        .KEY_XAB [OPEN$B_SIZ0])
                    THEN
                        XAB_STATUS = FOR$K_INVKEYSPE;

                    IF .KEY_XAB [OPEN$B_KTYPE] NEQ .KEY_XAB [XAB$B_DTP]
                    THEN
                        XAB_STATUS = FOR$K_INVKEYSPE;

                    BEGIN

                    LOCAL
                        NEXT;                       ! Address of next XAB in link
```

```
7                                 NEXT = .KEY_XAB [XAB$L_NXT];
7                                 FOR$$FREE_VM (OPEN$K_XAB_SIZE, .KEY_XAB);
7                                 KEY_XAB = .NEXT;
6                                 END;
6                                 KEY_COUNT = .KEY_COUNT - 3;
5                                 END;
5
4                             END;     ! Go through XABs
4
4                         !+
4                         ! If we had discovered any error while freeing the XAB's
4                         ! we report it now.  If we had reported it when we found it,
4                         ! we would have been left with some XABs laying around
4                         ! whose memory had not been deallocated.
4                         !-
4
4                         IF NOT .XAB_STATUS
4                         THEN
4                             $FOR$$SIGNAL_STO (.XAB_STATUS);
4
3                         END;          ! Indexed file
3
3                 END                                               ! End of old file processing
2             ELSE
2 !<BLF/PAGE>
```

```
!

!+
! Else (file was created)
!    Make sure V_APPEND is off so BACKSPACE will work.
!-

    BEGIN
    CCB [LUB$V_APPEND] = 0;
    END;

!+
! If this is not a disk or terminal, and if RECL was not specified,
! then reduce the default recordsize to fit within the blocksize.
!-

IF .V_DEFAULT_SIZE
THEN
    IF (NOT .BLOCK [FAB [FAB$L_DEV], DEV$V_RND;4, BYTE]) AND
        (NOT .BLOCK [FAB [FAB$L_DEV], DEV$V_TRM;4, BYTE]) AND
        (.FAB [FAB$W_BLS] NEQ 0)
    THEN
        BEGIN
        LOCAL
            NEW_RECL: WORD;
        NEW_RECL = .FAB [FAB$W_BLS];
        IF .CCB [LUB$V_SEGMENTED]
        THEN
            NEW_RECL = .NEW_RECL - 4;        ! Compensate for length
        IF .NEW_RECL LSSU .CCB [LUB$W_RBUF_SIZE]
        THEN
            CCB [LUB$W_RBUF_SIZE] = .NEW_RECL;
        END;

!+
! If this is a process-permanent file, ignore the carriage-control
! attributes RMS returned in the FAB and use the ones we set
! originally.  RMS will properly convert our writes anyway.
!-

IF .NAM [NAM$V_PPF]
THEN
    FAB [FAB$B_RAT] = .ORIG_RAT;

!+
! Set up the list-directed output record size as RECL, if specified,
! else 81 (80 if not FORTRAN carriage control).
!-

CCB [LUB$W_R_MARGIN] = (IF NOT .V_DEFAULT_SIZE THEN .CCB [LUB$W_RBUF_SIZE] ELSE
    (IF .FAB [FAB$V_FTN] THEN 81 ELSE 80));

!+
! Set bits in the LUB to indicate the file's carriage control
! characteristics.  This information is used by INQUIRE.
!-
```

```
2       IF .FAB [FAB$V_FTN]
2       THEN
2           CCB [LUB$V_FTN] = 1;
2       IF .FAB [FAB$V_CR]
2       THEN
2           CCB [LUB$V_CR] = 1;
2       IF .FAB [FAB$V_PRN]
2       THEN
2           CCB [LUB$V_PRN] = 1;
2
2       !+
2       ! Allocate record buffer dynamically from LUB$W_RBUF_SIZE setting in bytes.
2       ! Set LUB$A_RBUF_ADR to address of buffer allocated.
2       !-
2
2       CCB [LUB$A_RBUF_ADR] = FOR$$GET_VM (.CCB [LUB$W_RBUF_SIZE]);
2
2       !+
2       ! Allocate dynamic storage for the file name so the name can be
2       ! used later on for error diagnostics.  Point the LUB to the new
2       ! location.  (The size is already correct!)
2       ! Indicate that the string name is now stored in virtual memory so
2       ! it will be deallocated!
2       !-
2
3       BEGIN
3
3       LOCAL
3           T;
3
3       T = FOR$$GET_VM (.CCB [LUB$B_RSL]);
3       CH$MOVE (.CCB [LUB$B_RSL], .CCB [LUB$A_RSN], .T);
3       CCB [LUB$A_RSN] = .T;
3       NAM [NAM$L_RSA] = .T;
3       NAM [NAM$L_ESA] = .T;
3       NAM [NAM$B_ESL] = .CCB [LUB$B_RSL];
3       CCB [LUB$V_VIRT_RSN] = 1;
2       END;
2
2       !+
2       ! Store a code in the LUB indicating the type of organization.
2       !-
2
2       SELECTONE (.FAB [FAB$B_ORG]) OF
2           SET
2
2           [FAB$C_SEQ] :
2               CCB [LUB$B_ORGAN] = LUB$K_ORG_SEQUE;
2
2           [FAB$C_REL] :
2               CCB [LUB$B_ORGAN] = LUB$K_ORG_RELAT;
2
2           [FAB$C_IDX] :
3               BEGIN
3
3               IF .CCB [LUB$V_SEGMENTED] THEN $FOR$$SIGNAL_STO (FOR$K_INCRECTYP);
3
```

```
3                CCB [LUB$B_ORGAN] = LUB$K_ORG_INDEX;
2                END;
2
2         [OTHERWISE] :
2             $FOR$$SIGNAL_STO (FOR$K_INCFILORG);
2         TES;
2
2     !+
2     ! Set RAB fields that seldom change: UBF and USZ
2     !-
2
2     CCB [RAB$L_UBF] = .CCB [LUB$A_RBUF_ADR];
2     CCB [RAB$W_USZ] = .CCB [LUB$W_RBUF_SIZE];
2     CCB [LUB$A_UBF] = .CCB [LUB$A_RBUF_ADR];
2
2     !+
2     ! If the file is a sequential organization, sequential access,
2     ! disk file which is not a PPF, enable RFA cacheing for BACKSPACE.
2     !-
2
2     IF NOT .CCB [LUB$V_NOTSEQORG] AND
2         NOT .CCB [LUB$V_DIRECT] AND
2         NOT .CCB [LUB$V_FIXED] AND
2         NOT .NAM [NAM$V_PPF] AND
2         NOT .FAB [FAB$V_SQO]
2     THEN
3         BEGIN
3         BIND
3             FAB_DEV = FAB [FAB$L_DEV]: BLOCK [4, BYTE];
3         IF .FAB_DEV [DEV$V_RND] ! Random-access device?
3         THEN
4             BEGIN
4             LOCAL
4                 RCE: REF RCE_R_RCE_STRUCT,
4                 OLD_RCE: REF RCE_R_RCE_STRUCT;
4
4             !+
4             ! Allocate space for the RFA cache entries.
4             !-
4
4             RCE = FOR$$GET_VM (
4                 (RCE_K_CACHE_SIZE * RCE_S_RCE_STRUCT));
4
4             !+
4             ! Create a circularly linked list of entries and zero the
4             ! LOG_RECNO field of each entry.
4             !-
4
4             CCB [LUB$A_RFA_CACHE_BEG] = .RCE;    ! First allocated byte
4             CCB [LUB$A_RFA_CACHE_PTR] = .RCE;    ! Current entry
4             OLD_RCE = .RCE + (RCE_K_CACHE_SIZE - 1) * RCE_S_RCE_STRUCT;
4             DECRU I FROM RCE_K_CACHE_SIZE TO 1 DO
5                 BEGIN
5                 OLD_RCE [RCE_A_NEXT] = .RCE;
5                 RCE [RCE_A_PREV] = .OLD_RCE;
5                 RCE [RCE_L_LOG_RECNO] = 0;
5                 OLD_RCE = .RCE;
```

```
; 1978          2019 5                  RCE = .RCE + RCE_S_RCE_STRUCT;
; 1979          2020 4                  END;
; 1980          2021 4
; 1981          2022 4              CCB [LUB$V_RFA_CACHE_ENABLE] = 1;
; 1982          2023 3              END;
; 1983          2024 2          END;
; 1984          2025 2
; 1985          2026 2  !+
; 1986          2027 2  ! Indicate that the file is now FORTRAN opened.
; 1987          2028 2  !-
; 1988          2029 2      CCB [LUB$B_LANGUAGE] = LUB$K_LANG_FOR;
; 1989          2030 2      CCB [LUB$V_OPENED] = 1;
; 1990          2031 2  !+
; 1991          2032 2  ! Make sure that the FORTRAN exit handler will be called when the image
; 1992          2033 2  ! exits to purge the file's I/O buffers and close it, if necessary.
; 1993          2034 2  !-
; 1994          2035 2
; 1995          2036 2      IF ( NOT .FOR$$L_XIT_LOCK) THEN FOR$$DECL_EXITH ();
; 1996          2037 2
; 1997          2038 2      RETURN;                                   ! Return from OPEN_PROC routine
; 1998          2039 1      END;                                      ! End of OPEN_PROC routine


               3A  54  55  50  4E  49  24  53  59  53  00025 P.AAA:  .ASCII  \SYS$INPUT:\                        ;
           3A  54  55  50  54  55  4F  24  53  59  53  0002F P.AAB:  .ASCII  \SYS$OUTPUT:\                       ;
           54  41  44  2E  44  41  45  52  52  4F  46  0003A P.AAC:  .ASCII  \FORREAD.DAT\                       ;
               3A  44  41  45  52  24  52  4F  46  00045 P.AAD:  .ASCII  \FOR$READ:\                         ;
   54  41  44  2E  54  50  45  43  43  41  52  4F  46  0004E P.AAE:  .ASCII  \FORACCEPT.DAT\                     ;
               3A  54  50  45  43  43  41  24  52  4F  46  0005B P.AAF:  .ASCII  \FOR$ACCEPT:\                       ;
           54  41  44  2E  45  50  59  54  52  4F  46  00066 P.AAG:  .ASCII  \FORTYPE.DAT\                       ;
               3A  45  50  59  54  24  52  4F  46  00071 P.AAH:  .ASCII  \FOR$TYPE:\                         ;
       54  41  44  2E  54  4E  49  52  50  52  4F  46  0007A P.AAI:  .ASCII  \FORPRINT.DAT\                      ;
               3A  54  4E  49  52  50  24  52  4F  46  00086 P.AAJ:  .ASCII  \FOR$PRINT:\                        ;

                                                     A_SYS$INPUT=            P.AAA
                                                     A_SYS$OUTPUT=           P.AAB
                                                             .EXTRN  SYS$TRNLOG, SYS$PARSE
                                                             .EXTRN  SYS$OPEN, SYS$CREATE
                                                             .EXTRN  SYS$CONNECT

                                         07FC 00000          .ENTRY  FOR$$OPEN_PROC, Save R2,R3,R4,R5,R6,R7,R8,-  ; 0291
                                                                     R9,R10                                       ;
                        5E       FD88   CE  9E 00002         MOVAB   -632(SP), SP                                 ;
                                   E8   AB  D5 00007         TSTL    -24(CCB)                                     ; 0406
                                        3C  13 0000A         BEQL    1$                                           ;
                        56         E8   AB  D0 0000C         MOVL    -24(CCB), HEAP_FAB                           ; 0411
                        50         01   A6  9A 00010         MOVZBL  1(HEAP_FAB), R0                              ; 0412
          44   AB                  66       50  28 00014     MOVC3   R0, (HEAP_FAB), 68(CCB)                      ;
                                        56  DD 00019         PUSHL   HEAP_FAB                                     ; 0413
                        7E         01   A6  9A 0001B         MOVZBL  1(HEAP_FAB), -(SP)                           ;
              00000000G 00              02  FB 0001F         CALLS   #2, FOR$$FREE_VM                             ;
                                   E8   AB  D4 00026         CLRL    -24(CCB)                                     ; 0414
                        56         78   AB  9A 00029         MOVZBL  120(CCB), R6                                 ; 0415
                                        19  13 0002D         BEQL    1$                                           ;
     FEC8   CD        70   BB           56  28 0002F         MOVC3   R6, @112(CCB), TEMP_FNS                      ; 0418
                                   70   AB  DD 00036         PUSHL   112(CCB)                                     ; 0419
```

```
                                          56  DD 00039          PUSHL   R6                                      :
                  0000000G  00            02  FB 0003B          CALLS   #2, FOR$$FREE_VM                        :
                        70  AB      FEC8  CD  9E 00042          MOVAB   TEMP_FNS, 112(CCB)                      : 0420
                        59  CB      0094  CB  9E 00048 1$:      MOVAB   148(R11), R9                            : 0428
                        6C  AB            59  D0 0004D          MOVL    R9, 108(CCB)                            :
                        24  AE      0098  CB  9E 00051          MOVAB   152(CCB), 36(SP)                        : 0429
                        20  AE      00A0  CB  9E 00057          MOVAB   160(CCB), 32(SP)                        :
                        50            40  AE  9E 0005D          MOVAB   RES_OR_EXP_NAME, R0                     :
                        20  BE            50  D0 00061          MOVL    R0, @32(SP)                             :
                        24  BE            50  D0 00065          MOVL    R0, @36(SP)                             :
                      009E  CB            01  8E 00069          MNEGB   #1, 158(CCB)                            : 0430
                      0096  CB            01  8E 0006E          MNEGB   #1, 150(CCB)                            :
      2C            00      6E            00  2C 00073          MOVC5   #0, (SP), #0, #44, $RMS_PTR             : 0431
                                      C8  AD     00078                                                          :
                        C8  AD      2C1D  8F  B0 0007A          MOVW    #11293, $RMS_PTR                        :
                        68  AB        C8  AD  9E 00080          MOVAB   XAB_BLOCK, 104(CCB)                     : 0432
                        58            C8  AD  9E 00085          MOVAB   XAB_BLOCK, KEY_XAB                      : 0433
                        0C  AE        48  AB  9E 00089          MOVAB   72(CCB), 12(SP)                         : 0440
                        0C  BE            20  88 0008E          BISB2   #32, @12(SP)                            :
                                          55  D4 00092          CLRL    A_DEF_LOGNAM                            : 0469
                        56            79  AB  9E 00094          MOVAB   121(CCB), R6                            : 0544
                        52            74  AB  9E 00098          MOVAB   116(CCB), R2                            : 0545
                        53            70  AB  9E 0009C          MOVAB   112(CCB), R3                            : 0580
                03  FFFC  8F          C6  AB  AF 000A0          CASEW   -58(CCB), #-4, #3                       : 0471
    0103          00F0        00D5      00C2     000A7 2$:      .WORD   10$-2$,-                                :
                                                                        11$-2$,-                                :
                                                                        13$-2$,-                                :
                                                                        14$-2$                                  :
                        50            04  AC  D0 000AF          MOVL    OPEN_ADR, R0                            : 0516
                        57            38  A0  D0 000B3          MOVL    56(R0), R7                              :
                                          05  13 000B7          BEQL    3$                                      :
                                      68  A0  D5 000B9          TSTL    104(R0)                                 : 0517
                                          56  12 000BC          BNEQ    4$                                      :
                        F4  AD      4F46  8F  B0 000BE 3$:      MOVW    #20294, T_DFLT_FILE_NAM                 : 0520
                        F6  AD        52  8F  90 000C4          MOVB    #82, T_DFLT_FILE_NAM+2                  : 0522
                        51            C6  AB  32 000C9          CVTWL   -58(CCB), R1                            : 0523
                        51  00000064  8F  C6 000CD              DIVL2   #100, R1                                :
      7E            00      51            01  7A 000D4          EMUL    #1, R1, #0, -(SP)                       :
      51            51      8E            0A  7B 000D9          EDIV    #10, (SP)+, R1, R1                      :
                F7  AD      51            30  81 000DE          ADDB3   #48, R1, T_DFLT_FILE_NAM+3              :
                            51        C6  AB  32 000E3          CVTWL   -58(CCB), R1                            : 0524
                            51            0A  C6 000E7          DIVL2   #10, R1                                 :
      7E            00      51            01  7A 000EA          EMUL    #1, R1, #0, -(SP)                       :
      51            51      8E            0A  7B 000EF          EDIV    #10, (SP)+, R1, R1                      :
                F8  AD      51            30  81 000F4          ADDB3   #48, R1, T_DFLT_FILE_NAM+4              :
                            51        C6  AB  32 000F9          CVTWL   -58(CCB), R1                            : 0525
      7E            00      51            01  7A 000FD          EMUL    #1, R1, #0, -(SP)                       :
      51            51      8E            0A  7B 00102          EDIV    #10, (SP)+, R1, R1                      :
                F9  AD      51            30  81 00107          ADDB3   #48, R1, T_DFLT_FILE_NAM+5              :
                        FA  AD 5441442E   8F  D0 0010C          MOVL    #1413563438, T_DFLT_FILE_NAM+6          : 0526
                                      68  A0  D5 00114 4$:      TSTL    104(R0)                                 : 0537
                                          14  13 00117          BEQL    5$                                      :
                        54            68  A0  D0 00119          MOVL    104(R0), NAM_DSC                        : 0542
                      00FF  8F            64  B1 0011D          CMPW    (NAM_DSC), #255                         : 0543
                                          1C  1A 00122          BGTRU   7$                                      :
                        66                64  90 00124          MOVB    (NAM_DSC), (R6)                         : 0544
                        62            04  A4  D0 00127          MOVL    4(NAM_DSC), (R2)                        : 0545
```

```
                           07  11 0012B          BRB     6$                                        ; 0537
          66               0A  90 0012D 5$:      MOVB    #10, (R6)                                 ; 0554
          62         F4    AD  9E 00130          MOVAB   T_DFLT_FILE_NAM, (R2)                     ; 0555
                           57  D5 00134 6$:      TSTL    R7                                        ; 0563
                           17  13 00136          BEQL    9$                                        ;
          52               57  D0 00138          MOVL    R7, NAM_DSC                               ; 0575
    00FF  8F               62  B1 0013B          CMPW    (NAM_DSC), #255                           ; 0577
                           03  1B 00140 7$:      BLEQU   8$                                        ;
                         06AD  31 00142          BRW     139$                                      ;
      78  AB               62  90 00145 8$:      MOVB    (NAM_DSC), 120(CCB)                       ; 0579
          63         04    A2  D0 00149          MOVL    4(NAM_DSC), (R3)                          ; 0580
                           74  11 0014D          BRB     16$                                       ; 0563
                           63  D5 0014F 9$:      TSTL    (R3)                                      ; 0592
                           70  12 00151          BNEQ    16$                                       ;
      78  AB               06  90 00153          MOVB    #6, 120(CCB)                              ; 0595
          63         F4    AD  9E 00157          MOVAB   T_DFLT_FILE_NAM, (R3)                     ; 0596
          05         C6    AB  B1 0015B          CMPW    -58(CCB), #5                              ; 0604
                           2C  13 0015F          BEQL    12$                                       ;
          06         C6    AB  B1 00161          CMPW    -58(CCB), #6                              ; 0612
                           5C  12 00165          BNEQ    16$                                       ;
                           52  11 00167          BRB     15$                                       ; 0615
          66               0B  90 00169 10$:     MOVB    #11, (R6)                                 ; 0476
          62       FE3A    CF  9E 0016C          MOVAB   P.AAC, (R2)                               ; 0477
      78  AB               09  90 00171          MOVB    #9, 120(CCB)                              ; 0478
          63       FE3C    CF  9E 00175          MOVAB   P.AAD, (R3)                               ; 0479
                           11  11 0017A          BRB     12$                                       ; 0480
          66               0D  90 0017C 11$:     MOVB    #13, (R6)                                 ; 0486
          62       FE3B    CF  9E 0017F          MOVAB   P.AAE, (R2)                               ; 0487
      78  AB               0B  90 00184          MOVB    #11, 120(CCB)                             ; 0488
          63       FE3F    CF  9E 00188          MOVAB   P.AAF, (R3)                               ; 0489
          55       FE04    CF  9E 0018D 12$:     MOVAB   A_SYS$INPUT, A_DEF_LOGNAM                 ; 0490
          54               0A  D0 00192          MOVL    #10, L_DEF_LOGNAM                         ; 0491
                           2C  11 00195          BRB     16$                                       ; 0471
          66               0B  90 00197 13$:     MOVB    #11, (R6)                                 ; 0496
          62       FE38    CF  9E 0019A          MOVAB   P.AAG, (R2)                               ; 0497
      78  AB               09  90 0019F          MOVB    #9, 120(CCB)                              ; 0498
          63       FE3A    CF  9E 001A3          MOVAB   P.AAH, (R3)                               ; 0499
                           11  11 001A8          BRB     15$                                       ; 0500
          66               0C  90 001AA 14$:     MOVB    #12, (R6)                                 ; 0506
          62       FE39    CF  9E 001AD          MOVAB   P.AAI, (R2)                               ; 0507
      78  AB               0A  90 001B2          MOVB    #10, 120(CCB)                             ; 0508
          63       FE3C    CF  9E 001B6          MOVAB   P.AAJ, (R3)                               ; 0509
          55       FDE0    CF  9E 001BB 15$:     MOVAB   A_SYS$OUTPUT, A_DEF_LOGNAM                ; 0510
          54               0B  D0 001C0          MOVL    #11, L_DEF_LOGNAM                         ; 0511
                           55  D5 001C3 16$:     TSTL    A_DEF_LOGNAM                              ; 0631
                           47  13 001C5          BEQL    18$                                       ;
      3A  AE       010E    8F  B0 001C7          MOVW    #270, LOGNAM_DSC+2                        ; 0640
      30  AE   010E00FF    8F  D0 001CD          MOVL    #17694975, RESULT_DSC                     ; 0643
      34  AE         40    AE  9E 001D5          MOVAB   RES_OR_EXP_NAME, RESULT_DSC+4             ; 0644
      3C  AE               63  D0 001DA          MOVL    (R3), LOGNAM_DSC+4                        ; 0645
      38  AE         78    AB  9B 001DE          MOVZBW  120(CCB), LOGNAM_DSC                      ; 0646
                     C6    AB  B5 001E3          TSTW    -58(CCB)                                  ; 0648
                           03  18 001E6          BGEQ    17$                                       ;
                     38    AE  B7 001E8          DECW    LOGNAM_DSC                                ; 0655
                           7E  7C 001EB 17$:     CLRQ    -(SP)                                     ; 0664
                           7E  D4 001ED          CLRL    -(SP)                                     ;
                     3C    AE  9F 001EF          PUSHAB  RESULT_DSC                                ;
```

```
                                  7E  D4 001F2        CLRL    -(SP)                                   ;
                        4C        AE  9F 001F4        PUSHAB  LOGNAM_DSC                              ;
        00000000G  00             06  FB 001F7        CALLS   #6, SYS$TRNLOG                          ;
        00000629   8F             50  D1 001FE        CMPL    R0, #1577                               ;
                                  07  12 00205        BNEQ    18$                                     ;
                   63             55  D0 00207        MOVL    A_DEF_LOGNAM, (R3)                      ; 0667
              78   AB             54  90 0020A        MOVB    L_DEF_LOGNAM, 120(CCB)                  ; 0668
              14   AE     F8      AB  9E 0020E 18$:   MOVAB   -8(CCB), 20(SP)                         ; 0679
              14   BE             63  D0 00213        MOVL    (R3), @20(SP)                           ;
              1C   AE     F7      AB  9E 00217        MOVAB   -9(CCB), 28(SP)                         ; 0680
              1C   BE     78      AB  90 0021C        MOVB    120(CCB), @28(SP)                       ;
                          6C      AB  D4 00221        CLRL    108(CCB)                                ; 0688
              08   AE     44      AB  9E 00224        MOVAB   68(CCB), 8(SP)                          ; 0689
                          08      AE  DD 00229        PUSHL   8(SP)                                   ;
        00000000G  00             01  FB 0022C        CALLS   #1, SYS$PARSE                           ;
                   0B             50  E9 00233        BLBC    R0, 19$                                 ;
       05    0085  CB             05  E1 00236        BBC     #5, 133(CCB), 19$                       ; 0694
              0C   BE     40      8F  88 0023C        BISB2   #64, @12(SP)                            ; 0696
                          4C      AB  D4 00241 19$:   CLRL    76(CCB)                                 ; 0698
              6C   AB             59  D0 00244        MOVL    R9, 108(CCB)                            ; 0699
                   57     04      AC  D0 00248        MOVL    OPEN_ADR, R7                            ; 0708
                   06     20      A7  E9 0024C        BLBC    32(R7), 20$                             ;
              FC   AB             04  88 00250        BISB2   #4, -4(CCB)                             ; 0711
                                  09  11 00254        BRB     21$                                     ; 0708
                          5A      AB  95 00256 20$:   TSTB    90(CCB)                                 ; 0715
                                  04  12 00259        BNEQ    21$                                     ;
              5A   AB             1F  90 0025B        MOVB    #31, 90(CCB)                            ; 0717
                          E0      AB  D5 0025F 21$:   TSTL    -32(CCB)                                ; 0728
                                  04  12 00262        BNEQ    22$                                     ;
              E0   AB             01  D0 00264        MOVL    #1, -32(CCB)                            ; 0730
              0C   BE   0400      8F  A8 00268 22$:   BISW2   #1024, @12(SP)                          ; 0732
       04          00     10      A7  CF 0026E        CASEL   16(R7), #0, #4                          ; 0734
002E        0027      000C      0027     00273 23$:   .WORD   25$-23$,-                               ;
                                0046     0027B                24$-23$,-                               ;
                                                              25$-23$,-                               ;
                                                              26$-23$,-                               ;
                                                              28$-23$                                 ;
                                  5A  11 0027D        BRB     31$                                     ; 0773
              FC   AB             10  88 0027F 24$:   BISB2   #16, -4(CCB)                            ; 0739
              0C   BE     40      8F  8A 00283        BICB2   #64, @12(SP)                            ; 0740
              1E   AB             01  90 00288        MOVB    #1, 30(CCB)                             ; 0741
              30   AB     E0      AB  9E 0028C        MOVAB   -32(CCB), 48(CCB)                       ; 0742
                          34      AB  94 00291        CLRB    52(CCB)                                 ; 0743
              04   AB             10  88 00294        BISB2   #16, 4(CCB)                             ; 0744
                                  30  11 00298        BRB     29$                                     ; 0734
              FD   AB     40      8F  88 0029A 25$:   BISB2   #64, -3(CCB)                            ; 0749
                                  13  11 0029F        BRB     27$                                     ; 0750
       55     FC   AB             02  E0 002A1 26$:   BBS     #2, -4(CCB), 36$                        ; 0755
              05   AB             01  88 002A6        BISB2   #1, 5(CCB)                              ; 0758
              FD   AB             20  88 002AA        BISB2   #32, -3(CCB)                            ; 0759
              0C   BE   0400      8F  AA 002AE        BICW2   #1024, @12(SP)                          ; 0760
                          1E      AB  94 002B4 27$:   CLRB    30(CCB)                                 ; 0761
                                  11  11 002B7        BRB     29$                                     ; 0734
              0C   BE     40      8F  8A 002B9 28$:   BICB2   #64, @12(SP)                            ; 0766
              1E   AB             01  90 002BE        MOVB    #1, 30(CCB)                             ; 0767
                          35      AB  94 002C2        CLRB    53(CCB)                                 ; 0768
              FD   AB     80      8F  88 002C5        BISB2   #128, -3(CCB)                           ; 0769
```

```
                04          00     3C  A7  CF 002CA 29$:     CASEL   60(R7), #0, #4                                  ; 0783
   001A       0013        000D       0013     002CF 30$:     .WORD   33$-30$,-                                       ;
                                     002F     002D7                  32$-30$,-                                       ;
                                                                     33$-30$,-                                       ;
                                                                     34$-30$,-                                       ;
                                                                     37$-30$                                         ;
                                     02EA  31 002D9 31$:     BRW     101$                                            ; 0818
                           FC   AB     08  88 002DC 32$:     BISB2   #8, -4(CCB)                                     ; 0787
                                       27  11 002E0          BRB     38$                                             ;
                22         FC   AB     03  E0 002E2 33$:     BBS     #3, -4(CCB), 38$                                ; 0791
                                       08  11 002E7          BRB     35$                                             ; 0793
                           FC   AB     20  88 002E9 34$:     BISB2   #32, -4(CCB)                                    ; 0801
                           0C   BE     10  88 002ED          BISB2   #16, @12(SP)                                    ; 0802
                05         FC   AB     02  E0 002F1 35$:     BBS     #2, -4(CCB), 36$                                ; 0803
                0E         FD   AB     05  E1 002F6          BBC     #5, -3(CCB), 38$                                ; 0804
                                     01C5  31 002FB 36$:     BRW     84$                                             ; 0806
 0C   BE        01              19     01  F0 002FE 37$:     INSV    #1, #25, #1, @12(SP)                            ; 0811
                F2         FC   AB     02  E0 00304          BBS     #2, -4(CCB), 36$                                ; 0812
                                52     08  A7  D0 00309 38$:     MOVL    8(R7), R2                                   ; 0831
                                53     01  D0 0030D          MOVL    #1, R3                                          ;
                                       52  D5 00310          TSTL    R2                                              ; 0834
                                       02  12 00312          BNEQ    39$                                             ;
                                       53  D4 00314          CLRL    R3                                              ;
                                01     52  D1 00316 39$:     CMPL    R2, #1                                          ; 0837
                                       07  12 00319          BNEQ    40$                                             ;
                                       53  D4 0031B          CLRL    R3                                              ;
                D9         FC   AB     05  E0 0031D          BBS     #5, -4(CCB), 36$                                ; 0839
                                02     52  D1 00322 40$:     CMPL    R2, #2                                          ; 0841
                                       0A  13 00325          BEQL    41$                                             ;
                                05     52  D1 00327          CMPL    R2, #5                                          ;
                                       11  19 0032A          BLSS    42$                                             ;
                                06     52  D1 0032C          CMPL    R2, #6                                          ;
                                       0C  14 0032F          BGTR    42$                                             ;
                                       53  D4 00331 41$:     CLRL    R3                                              ;
                C3         FC   AB     02  E0 00333          BBS     #2, -4(CCB), 36$                                ; 0844
                           FC   AB  40 8F  88 00338          BISB2   #64, -4(CCB)                                    ; 0847
                                03     52  D1 0033D 42$:     CMPL    R2, #3                                          ; 0850
                                       05  13 00340          BEQL    43$                                             ;
                                05     52  D1 00342          CMPL    R2, #5                                          ;
                                       0C  12 00345          BNEQ    44$                                             ;
                                       53  D4 00347 43$:     CLRL    R3                                              ;
                AD         FC   AB     05  E0 00349          BBS     #5, -4(CCB), 36$                                ; 0854
                           FC   AB  80 8F  88 0034E          BISB2   #128, -4(CCB)                                   ; 0856
                                04     52  D1 00353 44$:     CMPL    R2, #4                                          ; 0859
                                       05  13 00356          BEQL    45$                                             ;
                                06     52  D1 00358          CMPL    R2, #6                                          ;
                                       0B  12 0035B          BNEQ    46$                                             ;
                                       53  D4 0035D 45$:     CLRL    R3                                              ;
                97         FC   AB     05  E0 0035F          BBS     #5, -4(CCB), 36$                                ; 0863
                           FF   AB     20  88 00364          BISB2   #32, -1(CCB)                                    ; 0867
                                3A     53  E8 00368 46$:     BLBS    R3, 53$                                         ; 0871
                03 FFFFFFFF     8F  14 A7  CF 0036B          CASEL   20(R7), #-1, #3                                 ; 0882
   001A       0014        000A       001E     00374 47$:     .WORD   51$-47$,-                                       ;
                                                                     48$-47$,-                                       ;
                                                                     49$-47$,-                                       ;
                                                                     50$-47$                                         ;
                                       71  11 0037C          BRB     64$                                             ; 0903
```

```
            0B       FC  AB        04  E0 0037E 48$:   BBS     #4, -4(CCB), 50$                    ; 0890
                              FD   AB  95 00383        TSTB    -3(CCB)                             ;
                                   06  19 00386        BLSS    50$                                 ;
                     FD  AB        01  88 00388 49$:   BISB2   #1, -3(CCB)                         ; 0897
                                   04  11 0038C        BRB     51$                                 ;
                     FD  AB        02  88 0038E 50$:   BISB2   #2, -3(CCB)                         ; 0900
            06           00   50   A7  CF 00392 51$:   CASEL   80(R7), #0, #6                      ; 0913
  0020    001A         0010      004B     00397 52$:   .WORD   62$-52$,-                           ;
          0047         0041      003B     0039F                54$-52$,-                           ;
                                                               55$-52$,-                           ;
                                                               56$-52$,-                           ;
                                                               59$-52$,-                           ;
                                                               60$-52$,-                           ;
                                                               61$-52$                             ;
                                   76  11 003A5 53$:   BRB     70$                                 ; 0964
                     FD  AB        04  88 003A7 54$:   BISB2   #4, -3(CCB)                         ; 0928
                     63  AB        01  90 003AB        MOVB    #1, 99(CCB)                         ; 0929
                                   31  11 003AF        BRB     62$                                 ; 0913
                     63  AB        02  90 003B1 55$:   MOVB    #2, 99(CCB)                         ; 0934
                                   2B  11 003B5        BRB     62$                                 ; 0913
            03       FC  AB        04  E1 003B7 56$:   BBC     #4, -4(CCB), 58$                    ; 0940
                                 0104  31 003BC 57$:   BRW     84$                                 ;
                              FD   AB  95 003BF 58$:   TSTB    -3(CCB)                             ;
                                   F8  19 003C2        BLSS    57$                                 ;
                         F4   FD   AB  E8 003C4        BLBS    -3(CCB), 57$                        ;
                     63  AB        02  90 003C8        MOVB    #2, 99(CCB)                         ; 0944
                     FD  AB        08  88 003CC        BISB2   #8, -3(CCB)                         ; 0945
                                   10  11 003D0        BRB     62$                                 ; 0913
                     63  AB        04  90 003D2 59$:   MOVB    #4, 99(CCB)                         ; 0950
                                   0A  11 003D6        BRB     62$                                 ; 0913
                     63  AB        06  90 003D8 60$:   MOVB    #6, 99(CCB)                         ; 0955
                                   04  11 003DC        BRB     62$                                 ; 0913
                     63  AB        05  90 003DE 61$:   MOVB    #5, 99(CCB)                         ; 0960
            03           00   1C   A7  CF 003E2 62$:   CASEL   28(R7), #0, #3                      ; 0974
  0018    0014         000E      000A     003E7 63$:   .WORD   65$-63$,-                           ;
                                                               66$-63$,-                           ;
                                                               67$-63$,-                           ;
                                                               68$-63$                             ;
                                   2C  11 003EF 64$:   BRB     70$                                 ; 0991
                         0A   FD   AB  E9 003F1 65$:   BLBC    -3(CCB), 68$                        ; 0979
                     62  AB        01  88 003F5 66$:   BISB2   #1, 98(CCB)                         ; 0982
                                   04  11 003F9        BRB     68$                                 ;
                     62  AB        02  88 003FB 67$:   BISB2   #2, 98(CCB)                         ; 0985
                         6E   62   AB  9E 003FF 68$:   MOVAB   98(CCB), (SP)                       ; 0999
                     28  AE   00   BE  90 00403        MOVB    @0(SP), ORIG_RAT                    ;
                         59   A0   AB  9E 00408        MOVAB   -96(CCB), R9                        ; 1008
                     01  A9        08  88 0040C        BISB2   #8, 1(R9)                           ;
            03           00   4C   A7  CF 00410        CASEL   76(R7), #0, #3                      ; 1010
  0039    0029         000B      000B     00415 69$:   .WORD   71$-69$,-                           ;
                                                               71$-69$,-                           ;
                                                               74$-69$,-                           ;
                                                               75$-69$                             ;
                                 01A6  31 0041D 70$:   BRW     101$                                ; 1047
            06       FC  AB        04  E1 00420 71$:   BBC     #4, -4(CCB), 72$                    ; 1016
                         02   63   AB  91 00425        CMPB    99(CCB), #2                         ;
                                   91  13 00429        BEQL    57$                                 ;
                              FD   AB  95 0042B 72$:   TSTB    -3(CCB)                             ; 1020
```

```
                                        05  18 0042E           BGEQ    73$                                 ;
                                   4C   A7  D5 00430           TSTL    76(R7)                              ;
                                        87  12 00433           BNEQ    57$                                 ;
                                   61   AB  94 00435 73$:      CLRB    97(CCB)                             ; 1024
                  01       A9           08  8A 00438           BICB2   #8, 1(R9)                           ; 1025
                                        23  11 0043C           BRB     77$                                 ; 1010
          15      FD       AB           03  E0 0043E 74$:      BBS     #3, -3(CCB), 76$                    ; 1031
                                   FD   AB  95 00443           TSTB    -3(CCB)                             ;
                                        7B  19 00446           BLSS    84$                                 ;
                  61       AB           10  90 00448           MOVB    #16, 97(CCB)                        ; 1033
                                        13  11 0044C           BRB     77$                                 ; 1010
          70      FC       AB           04  E0 0044E 75$:      BBS     #4, -4(CCB), 84$                    ; 1039
          6B      FD       AB           05  E0 00453           BBS     #5, -3(CCB), 84$                    ;
          66      FD       AB           03  E0 00458 76$:      BBS     #3, -3(CCB), 84$                    ;
                  61       AB           20  90 0045D           MOVB    #32, 97(CCB)                        ; 1043
          0B               69           0B  E1 00461 77$:      BBC     #11, (R9), 78$                      ; 1054
          50  0E000000     8F      63   AB  78 00465           ASHL    99(CCB), #234881024, R0             ; 1055
                                        53  19 0046E           BLSS    84$                                 ;
                                   50   A7  D5 00470 78$:      TSTL    80(R7)                              ; 1065
                                        2D  12 00473           BNEQ    81$                                 ;
                           10      61   AB  91 00475           CMPB    97(CCB), #16                        ; 1068
                                        10  13 00479           BEQL    79$                                 ;
                           20      61   AB  91 0047B           CMPB    97(CCB), #32                        ;
                                        0A  13 0047F           BEQL    79$                                 ;
          05      FC       AB           04  E0 00481           BBS     #4, -4(CCB), 79$                    ;
                                   FD   AB  95 00486           TSTB    -3(CCB)                             ; 1069
                                        0A  18 00489           BGEQ    80$                                 ;
                  63       AB           01  90 0048B 79$:      MOVB    #1, 99(CCB)                         ; 1072
                  FD       AB           04  88 0048F           BISB2   #4, -3(CCB)                         ; 1073
                                        0D  11 00493           BRB     81$                                 ; 1068
                  63       AB           02  90 00495 80$:      MOVB    #2, 99(CCB)                         ; 1077
          04      FD       AB           01  E1 00499           BBC     #1, -3(CCB), 81$                    ; 1079
                  FD       AB           08  88 0049E           BISB2   #8, -3(CCB)                         ;
                           0D      34   A7  E9 004A2 81$:      BLBC    52(R7), 82$                         ; 1089
                  5B       AB           0F  90 004A6           MOVB    #15, 91(CCB)                        ; 1092
          05               69           0B  E0 004AA           BBS     #11, (R9), 82$                      ; 1094
                  5B       AB      40   8F  88 004AE           BISB2   #64, 91(CCB)                        ; 1096
                  18       AE      5C   A7  D0 004B3 82$:      MOVL    92(R7), 24(SP)                      ; 1107
                                        03  12 004B8           BNEQ    83$                                 ;
                                      00FE  31 004BA           BRW     99$                                 ;
                           20      61   AB  91 004BD 83$:      CMPB    97(CCB), #32                        ; 1117
                                        05  13 004C1           BEQL    85$                                 ;
                                        2E  DD 004C3 84$:      PUSHL   #46                                 ;
                                      065A  31 004C5           BRW     214$                                ;
                           56      18   AE  D0 004C8 85$:      MOVL    24(SP), KEY_DEFN                    ; 1119
                  04       AE      02   A6  32 004CC           CVTWL   2(KEY_DEFN), KEY_COUNT              ; 1120
                           56           04  C0 004D1           ADDL2   #4, KEY_DEFN                        ; 1121
    7E            00  04   AE           01  7A 004D4           EMUL    #1, KEY_COUNT, #0, -(SP)            ; 1123
    50            50       8E           03  7B 004DA           EDIV    #3, (SP)+, R0, R0                   ;
                                        50  D5 004DF           TSTL    R0                                  ;
                                        03  13 004E1           BEQL    86$                                 ;
                                      00E0  31 004E3           BRW     101$                                ;
                  04       AE           03  C6 004E6 86$:      DIVL2   #3, KEY_COUNT                       ; 1125
                           5A           01  CE 004EA           MNEGL   #1, KEY_NUM                         ; 1131
                                      00C1  31 004ED           BRW     97$                                 ;
                           7E      50   8F  9A 004F0 87$:      MOVZBL  #80, -(SP)                          ; 1133
              00000000G    00           01  FB 004F4           CALLS   #1, FOR$$GET_VM                     ;
```

```
                                                                  M 12
FOR$$OPEN_DEFLT FORTRAN default open                              16-Sep-1984 00:37:00    VAX-11 Bliss-32 V4.0-742            Page  53
1-098                                                             14-Sep-1984 12:32:16    [FORRTL.SRC]FOROPENDE.B32;1               (25)

                           10    AE                 50  D0 004FB         MOVL    R0, XAB_ADDR                              :
                           04    A8          10     AE  D0 004FF         MOVL    XAB_ADDR, 4(KEY_XAB)                      : 1134
                                 58          10     AE  D0 00504         MOVL    XAB_ADDR, KEY_XAB                         : 1135
    0050   8F        00          6E                 00  2C 00508         MOVC5   #0, (SP), #0, #80, (KEY_XAB)              : 1141
                                                    68     0050F
                                 68        4C15     8F  B0 00510         MOVW    #19477, (KEY_XAB)                         : 1142
                                 52          04     A6  D0 00515         MOVL    4(KEY_DEFN), R2                           : 1149
                                                    03  14 00519         BGTR    89$                                       :
                                                  02F3  31 0051B 88$:    BRW     141$                                      :
                    00007FFF     8F                 52  D1 0051E 89$:    CMPL    R2, #32767                                : 1150
                                                    F4  14 00525         BGTR    88$                                       :
                    00007FFF     8F          08     A6  D1 00527         CMPL    8(KEY_DEFN), #32767                       : 1151
                                                    EA  14 0052F         BGTR    88$                                       :
                                 52          08     A6  D1 00531         CMPL    8(KEY_DEFN), R2                           : 1152
                                                    E4  19 00535         BLSS    88$                                       :
                1E    A8         52                 01  A3 00537         SUBW3   #1, R2, 30(KEY_XAB)                       : 1156
                      52         08                 A6  52 C3 0053C      SUBL3   R2, 8(KEY_DEFN), R2                       : 1163
                                                    52  D6 00541         INCL    SIZE                                      :
                    000000FF     8F                 52  D1 00543         CMPL    SIZE, #255                                : 1165
                                                    CF  14 0054A         BGTR    88$                                       :
                           2E    A8                 52  90 0054C         MOVB    SIZE, 46(KEY_XAB)                         : 1167
                           4E    A8          1E     A8  B0 00550         MOVW    30(KEY_XAB), 78(KEY_XAB)                  : 1169
                           4D    A8          2E     A8  90 00555         MOVB    46(KEY_XAB), 77(KEY_XAB)                  : 1170
                                                    66  95 0055A         TSTB    (KEY_DEFN)                                : 1173
                                                    05  13 0055C         BEQL    90$                                       :
                                 0E                 66  91 0055E         CMPB    (KEY_DEFN), #14                           :
                                                    04  12 00561         BNEQ    91$                                       :
                                                    50  D4 00563 90$:    CLRL    R0                                        :
                                                    32  11 00565         BRB     95$                                       :
                                 03                 66  91 00567 91$:    CMPB    (KEY_DEFN), #3                            : 1174
                                                    15  13 0056A         BEQL    92$                                       :
                                 07                 66  91 0056C         CMPB    (KEY_DEFN), #7                            : 1175
                                                    25  13 0056F         BEQL    94$                                       :
                                 04                 66  91 00571         CMPB    (KEY_DEFN), #4                            : 1176
                                                    10  12 00574         BNEQ    93$                                       :
                                 04          2E     A8  91 00576         CMPB    46(KEY_XAB), #4                           :
                                                    05  12 0057A         BNEQ    92$                                       :
                                 50                 04  D0 0057C         MOVL    #4, R0                                    :
                                                    18  11 0057F         BRB     95$                                       :
                                 50                 02  D0 00581 92$:    MOVL    #2, R0                                    :
                                                    13  11 00584         BRB     95$                                       :
                                 08                 66  91 00586 93$:    CMPB    (KEY_DEFN), #8                            : 1177
                                                    3B  12 00589         BNEQ    101$                                      :
                                 04          2E     A8  91 0058B         CMPB    46(KEY_XAB), #4                           :
                                                    05  12 0058F         BNEQ    94$                                       :
                                 50                 03  D0 00591         MOVL    #3, R0                                    :
                                                    03  11 00594         BRB     95$                                       :
                                 50                 01  D0 00596 94$:    MOVL    #1, R0                                    :
                           13    A8                 50  90 00599 95$:    MOVB    R0, 19(KEY_XAB)                           : 1171
                           4C    A8          13     A8  90 0059D         MOVB    19(KEY_XAB), 76(KEY_XAB)                  : 1183
                                                    5A  D5 005A2         TSTL    KEY_NUM                                   : 1185
                                                    04  13 005A4         BEQL    96$                                       :
                           12    A8                 03  88 005A6         BISB2   #3, 18(KEY_XAB)                           : 1189
                           17    A8                 5A  90 005AA 96$:    MOVB    KEY_NUM, 23(KEY_XAB)                      : 1192
                                 56                 0C  C0 005AE         ADDL2   #12, KEY_DEFN                             : 1193
                      02         5A          04     AE  F2 005B1 97$:    AOBLSS  KEY_COUNT, KEY_NUM, 98$                   : 1131
                                                    03  11 005B6         BRB     99$                                       :
```

```
                                          FF35  31 005B8 98$:    BRW     87$                                     ;
          02           00           60      A7  CF 005BB 99$:    CASEL   96(R7), #0, #2                          ; 1204
        000B         0010                 0010     005C0 100$:   .WORD   103$-100$,-                             ;
                                                                         103$-100$,-                             ;
                                                                         102$-100$                               ;
                                            30  DD 005C6 101$:   PUSHL   #48                                     ; 1214
                                          0557  31 005C8         BRW     214$                                    ;
                         FF    AB           40  8F  88 005CB 102$: BISB2 #64, -1(CCB)                            ; 1211
                                            55  D4 005D0 103$:   CLRL    V_DEFAULT_SIZE                          ; 1230
                         50    18           A7  D0 005D2         MOVL    24(R7), R0                              ; 1232
                                            3C  12 005D6         BNEQ    110$                                    ; 1235
                                      D2    AB  B5 005D8         TSTW    -46(CCB)                                ; 1243
                                            6D  12 005DB         BNEQ    115$                                    ;
          0E           FC    AB             03  E0 005DD         BBS     #3, -4(CCB), 106$                       ; 1247
          03           FD    AB             02  E1 005E2         BBC     #2, -3(CCB), 105$                       ;
                                          03F6  31 005E7 104$:   BRW     188$                                    ;
                         10           61    AB  91 005EA 105$:   CMPB    97(CCB), #16                            ; 1248
                                            F7  13 005EE         BEQL    104$                                    ;
          12           FD    AB             01  E1 005F0 106$:   BBC     #1, -3(CCB), 108$                       ; 1253
          06           FD    AB             02  E1 005F5         BBC     #2, -3(CCB), 107$                       ; 1255
                             50       80    8F  9A 005FA         MOVZBL  #128, R0                                ;
                                            0B  11 005FE         BRB     109$                                    ;
                             50     07FC    8F  3C 00600 107$:   MOVZWL  #2044, R0                               ;
                                            04  11 00605         BRB     109$                                    ;
                             50       85    8F  9A 00607 108$:   MOVZBL  #133, R0                                ; 1253
                       D2    AB             50  B0 0060B 109$:   MOVW    R0, -46(CCB)                            ; 1252
                             55             01  D0 0060F         MOVL    #1, V_DEFAULT_SIZE                      ; 1262
                                            36  11 00612         BRB     115$                                    ; 1243
             00007FFF        8F             50  D1 00614 110$:   CMPL    R0, #32767                              ; 1265
                                            CA  1A 0061B         BGTRU   104$                                    ;
          05           FD    AB             01  E1 0061D         BBC     #1, -3(CCB), 111$                       ; 1271
                             51             04  D0 00622         MOVL    #4, R1                                  ;
                                            03  11 00625         BRB     112$                                    ;
                             51             01  D0 00627 111$:   MOVL    #1, R1                                  ;
                             51             50  C4 0062A 112$:   MULL2   R0, R1                                  ;
          05           FD    AB             03  E1 0062D         BBC     #3, -3(CCB), 113$                       ; 1272
                             50             02  D0 00632         MOVL    #2, R0                                  ;
                                            02  11 00635         BRB     114$                                    ;
                                            50  D4 00637 113$:   CLRL    R0                                      ;
          52                 51             50  C1 00639 114$:   ADDL3   R0, R1, T                               ;
             00007FFF        8F             52  D1 0063D         CMPL    T, #32767                               ; 1274
                                            A1  1A 00644         BGTRU   104$                                    ;
                       D2    AB             52  B0 00646         MOVW    T, -46(CCB)                             ; 1276
                             56       FC    AB  9E 0064A 115$:   MOVAB   -4(CCB), R6                             ; 1283
          0C                 66             0A  E0 0064E         BBS     #10, (R6), 116$                         ;
                             10       61    AB  91 00652         CMPB    97(CCB), #16                            ; 1284
                                            06  13 00656         BEQL    116$                                    ;
                             20       61    AB  91 00658         CMPB    97(CCB), #32                            ; 1285
                                            05  12 0065C         BNEQ    117$                                    ;
                       7A    AB       D2    AB  B0 0065E 116$:   MOVW    -46(CCB), 122(CCB)                      ; 1286
                                      28    A7  D5 00663 117$:   TSTL    40(R7)                                  ; 1296
                                            13  13 00666         BEQL    119$                                    ;
                             50       28    A7  D0 00668         MOVL    40(R7), R0                              ; 1299
                                            03  18 0066C         BGEQ    118$                                    ;
                             50             50  CE 0066E         MNEGL   R0, R0                                  ;
                       54    AB             50  D0 00671 118$:   MOVL    R0, 84(CCB)                             ;
   0C    BE    01            15             01  F0 00675         INSV    #1, #21, #1, @12(SP)                    ; 1300
```

```
                                    2C  A7  D5 0067B 119$:   TSTL    44(R7)                              ; 1308
                                        16  13 0067E         BEQL    121$                                ;
                          50        2C  A7  D0 00680         MOVL    44(R7), R0                          ; 1311
                                        03  18 00684         BGEQ    120$                                ;
                          50            50  CE 00686         MNEGL   R0, R0                              ;
                    00010000  8F        50  D1 00689 120$:   CMPL    R0, #65536                          ;
                                        65  1E 00690         BGEQU   124$                                ;
                          58  AB        50  B0 00692         MOVW    R0, 88(CCB)                         ; 1313
      00   BE          01      03   30  A7  F0 00696 121$:   INSV    48(R7), #3, #1, @0(SP)              ; 1321
                                    40  A7  D5 0069D         TSTL    64(R7)                              ; 1329
                                        05  13 006A0         BEQL    122$                                ;
                          E4  AB    40  A7  D0 006A2         MOVL    64(R7), -28(CCB)                    ;
                          7C  AB    E4  AB  D0 006A7 122$:   MOVL    -28(CCB), 124(CCB)                  ; 1331
                          50        48  A7  D0 006AC         MOVL    72(R7), R0                          ; 1341
                                        30  13 006B0         BEQL    123$                                ; 1344
                    0000FFFF  8F        50  D1 006B2         CMPL    R0, #65535                          ; 1347
                                        3C  1A 006B9         BGTRU   124$                                ;
                        0080  CB        50  B0 006BB         MOVW    R0, 128(CCB)                        ; 1349
                              50      01FF  C0  9E 006C0     MOVAB   511(R0), R0                         ; 1350
                       51     50  00000200  8F  C7 006C5     DIVL3   #512, R0, R1                        ;
                          37  AB        51  90 006CD         MOVB    R1, 55(CCB)                         ;
                              50      0082  CB  9E 006D1     MOVAB   130(CCB), R0                        ; 1351
                              60        37  AB  90 006D6     MOVB    55(CCB), (R0)                       ;
                              3F        60  91 006DA         CMPB    (R0), #63                           ; 1352
                                        03  1B 006DD         BLEQU   123$                                ;
                              60        3F  90 006DF         MOVB    #63, (R0)                           ; 1354
                              50    24  A7  D0 006E2 123$:   MOVL    36(R7), R0                          ; 1370
                                        14  13 006E6         BEQL    125$                                ; 1373
                    0000007F  8F        50  D1 006E8         CMPL    R0, #127                            ; 1376
                                        06  1A 006EF         BGTRU   124$                                ;
                          36  AB        50  90 006F1         MOVB    R0, 54(CCB)                         ; 1377
                                        05  11 006F5         BRB     125$                                ;
                                        2D  DD 006F7 124$:   PUSHL   #45                                 ; 1380
                                      0426  31 006F9         BRW     214$                                ;
                                    44  A7  D5 006FC 125$:   TSTL    68(R7)                              ; 1387
                                        0C  13 006FF         BEQL    126$                                ;
                          DC  AB    44  A7  D0 00701         MOVL    68(R7), -36(CCB)                    ; 1390
                              04        67  E9 00706         BLBC    (R7), 126$                          ; 1392
                          01  A6        10  88 00709         BISB2   #16, 1(R6)                          ;
                                    54  A7  D5 0070D 126$:   TSTL    84(R7)                              ; 1407
                                        17  13 00710         BEQL    127$                                ;
                          2C  AE    C6  AB  32 00712         CVTWL   -58(CCB), LOG_UNIT                  ; 1414
                          01  A9        04  88 00717         BISB2   #4, 1(R9)                           ; 1415
                                    2C  AE  9F 0071B         PUSHAB  LOG_UNIT                            ; 1416
                                        5B  DD 0071E         PUSHL   CCB                                 ; 1417
                                    10  AE  DD 00720         PUSHL   16(SP)                              ; 1416
                          54  B7        03  FB 00723         CALLS   #3, @84(R7)                         ;
                                        29  11 00727         BRB     130$                                ;
                       0C     66        03  E1 00729 127$:   BBC     #3, (R6), 128$                      ; 1432
                                    08  AE  DD 0072D         PUSHL   8(SP)                               ; 1434
                    00000000G  00       01  FB 00730         CALLS   #1, SYS$OPEN                        ;
                                        0A  11 00737         BRB     129$                                ;
                                    08  AE  DD 00739 128$:   PUSHL   8(SP)                               ; 1436
                    00000000G  00       01  FB 0073C         CALLS   #1, SYS$CREATE                      ;
                              52        50  D0 00743 129$:   MOVL    R0, OPEN_STATUS                     ;
                              0C        52  E9 00746         BLBC    OPEN_STATUS, 131$                   ; 1442
                                        5B  DD 00749         PUSHL   CCB                                 ;
```

```
                 00000000G  00              01  FB 0074B           CALLS   #1, SYS$CONNECT                          ;
                            52              50  D0 00752 130$:     MOVL    R0, OPEN_STATUS                          ;
                                      68    AB  D4 00755 131$:     CLRL    104(CCB)                                 ; 1453
           0D          0C   BE              19  E1 00758           BBC     #25, @12(SP), 132$                       ; 1461
                 00010619   8F        4C    AB  D1 0075D           CMPL    76(CCB), #67097                          ;
                                            03  13 00765           BEQL    132$                                     ;
                            66              08  88 00767           BISB2   #8, (R6)                                 ;
           0E          FE   AB              00  E5 0076A 132$:     BBCC    #0, -2(CCB), 133$                        ; 1467
                                      14    BE  DD 0076F           PUSHL   @20(SP)                                  ; 1469
                            7E        20    BE  9A 00772           MOVZBL  @32(SP), -(SP)                           ;
                 00000000G  00              02  FB 00776           CALLS   #2, FOR$$FREE_VM                         ;
                                    0097    CB  95 0077D 133$:     TSTB    151(CCB)                                 ; 1477
                                            0D  13 00781           BEQL    134$                                     ;
                       14   BE        24    BE  D0 00783           MOVL    @36(SP), @20(SP)                         ; 1480
                       1C   BE      0097    CB  90 00788           MOVB    151(CCB), @28(SP)                        ; 1481
                                            11  11 0078E           BRB     135$                                     ; 1477
                                    009F    CB  95 00790 134$:     TSTB    159(CCB)                                 ; 1485
                                            0B  13 00794           BEQL    135$                                     ;
                       14   BE        20    BE  D0 00796           MOVL    @32(SP), @20(SP)                         ; 1488
                       1C   BE      009F    CB  90 0079B           MOVB    159(CCB), @28(SP)                        ; 1489
                            03              52  E9 007A1 135$:     BLBC    OPEN_STATUS, 136$                        ; 1507
                                          00CB  31 007A4           BRW     150$                                     ;
                            50        4C    AB  D0 007A7 136$:     MOVL    76(CCB), R0                              ; 1566
                 00018292   8F              50  D1 007AB           CMPL    R0, #98962                               ;
                                            04  12 007B2           BNEQ    137$                                     ;
                                            1D  DD 007B4           PUSHL   #29                                      ;
                                            5B  11 007B6           BRB     142$                                     ;
                 000184C4   8F              50  D1 007B8 137$:     CMPL    R0, #99524                               ;
                                            04  12 007BF           BNEQ    138$                                     ;
                                            2A  DD 007C1           PUSHL   #42                                      ;
                                            4E  11 007C3           BRB     142$                                     ;
                 0001852C   8F              50  D1 007C5 138$:     CMPL    R0, #99628                               ;
                                            24  13 007CC           BEQL    139$                                     ;
                 000185F4   8F              50  D1 007CE           CMPL    R0, #99828                               ;
                                            1B  13 007D5           BEQL    139$                                     ;
                 000186D4   8F              50  D1 007D7           CMPL    R0, #100052                              ;
                                            12  13 007DE           BEQL    139$                                     ;
                 000186E4   8F              50  D1 007E0           CMPL    R0, #100068                              ;
                                            09  13 007E7           BEQL    139$                                     ;
                 000186FC   8F              50  D1 007E9           CMPL    R0, #100092                              ;
                                            04  12 007F0           BNEQ    140$                                     ;
                                            2B  DD 007F2 139$:     PUSHL   #43                                      ;
                                            79  11 007F4           BRB     149$                                     ;
                 000185FC   8F              50  D1 007F6 140$:     CMPL    R0, #99836                               ;
                                            12  13 007FD           BEQL    141$                                     ;
                 00018624   8F              50  D1 007FF           CMPL    R0, #99876                               ;
                                            09  13 00806           BEQL    141$                                     ;
                 000186BC   8F              50  D1 00808           CMPL    R0, #100028                              ;
                                            04  12 0080F           BNEQ    143$                                     ;
                                            31  DD 00811 141$:     PUSHL   #49                                      ;
                                            5A  11 00813 142$:     BRB     149$                                     ;
                 0001C00A   8F              50  D1 00815 143$:     CMPL    R0, #114698                              ;
                                            4F  12 0081C           BNEQ    148$                                     ;
                            53              50  D0 0081E           MOVL    R0, OLD_STS                              ;
                            52        50    AB  D0 00821           MOVL    80(CCB), OLD_STV                         ;
                                      08    AE  DD 00825           PUSHL   8(SP)                                    ;
                 00000000G  00              01  FB 00828           CALLS   #1, SYS$PARSE                            ;
```

```
                          34            50  E9 0082F          BLBC    R0, 146$                                   ;
                    4C    AB            53  D0 00832          MOVL    OLD_STS, 76(CCB)                           ;
                    50    AB            52  D0 00836          MOVL    OLD_STV, 80(CCB)                           ;
             26   0084    CB            05  E1 0083A          BBC     #5, 132(CCB), 146$                         ;
                                0080    CB  B5 00840          TSTW    128(CCB)                                   ;
                                        20  13 00844          BEQL    146$                                       ;
             05                 66      0A  E0 00846          BBS     #10, (R6), 144$                            ;
                                50      04  D0 0084A          MOVL    #4, R0                                     ;
                                        02  11 0084D          BRB     145$                                       ;
                                        50  D4 0084F 144$:    CLRL    R0                                         ;
                          51      D2    AB  3C 00851 145$:    MOVZWL  -46(CCB), R1                               ;
                          50            51  C0 00855          ADDL2   R1, R0                                     ;
    50      0080    CB    10            00  ED 00858          CMPZV   #0, #16, 128(CCB), R0                      ;
                                        05  1E 0085F          BGEQU   146$                                       ;
                          50            25  D0 00861          MOVL    #37, R0                                    ;
                                        03  11 00864          BRB     147$                                       ;
                          50            1E  D0 00866 146$:    MOVL    #30, R0                                    ;
                                        50  DD 00869 147$:    PUSHL   R0                                         ;
                                        02  11 0086B          BRB     149$                                       ;
                                        1E  DD 0086D 148$:    PUSHL   #30                                        ;
                                      02B0  31 0086F 149$:    BRW     214$                                       ;
             03                 66      03  E0 00872 150$:    BBS     #3, (R6), 151$                             ; 1576
                                      01D1  31 00876          BRW     199$                                       ;
                                  4C    A7  D5 00879 151$:    TSTL    76(R7)                                     ; 1587
                                        21  13 0087C          BEQL    157$                                       ;
             02           01      4C    A7  CF 0087E          CASEL   76(R7), #1, #2                             ; 1594
           0011         000C          0008     00883 152$:    .WORD   153$-152$,-                                ;
                                                                      154$-152$,-                                ;
                                                                      155$-152$                                  ;
                                        4A  11 00889          BRB     163$                                       ; 1601
                                        50  D4 0088B 153$:    CLRL    T                                          ; 1594
                                        08  11 0088D          BRB     156$                                       ;
                          50            10  D0 0088F 154$:    MOVL    #16, T                                     ;
                                        03  11 00892          BRB     156$                                       ;
                          50            20  D0 00894 155$:    MOVL    #32, T                                     ;
    50        61    AB    08            00  ED 00897 156$:    CMPZV   #0, #8, 97(CCB), T                         ; 1605
                                        14  12 0089D          BNEQ    159$                                       ;
                                        66  B5 0089F 157$:    TSTW    (R6)                                       ; 1614
                                        06  18 008A1          BGEQ    158$                                       ;
                          20      61    AB  91 008A3          CMPB    97(CCB), #32                               ;
                                        0A  12 008A7          BNEQ    159$                                       ;
             09           66            04  E1 008A9 158$:    BBC     #4, (R6), 160$                             ;
                          20      61    AB  91 008AD          CMPB    97(CCB), #32                               ; 1615
                                        03  12 008B1          BNEQ    160$                                       ;
                                      026A  31 008B3 159$:    BRW     213$                                       ;
                          54      61    AB  9E 008B6 160$:    MOVAB   97(CCB), R4                                ; 1623
                                        64  95 008BA          TSTB    (R4)                                       ;
                                        04  13 008BC          BEQL    161$                                       ;
                    01    A9            08  88 008BE          BISB2   #8, 1(R9)                                  ;
             06           00      50    A7  CF 008C2 161$:    CASEL   80(R7), #0, #6                             ; 1635
    0055   0049         0043          0011     008C7 162$:    .WORD   164$-162$,-                                ;
           006D         0067          0061     008CF                  167$-162$,-                                ;
                                                                      168$-162$,-                                ;
                                                                      169$-162$,-                                ;
                                                                      170$-162$,-                                ;
                                                                      171$-162$,-                                ;
                                                                      172$-162$                                  ;
```

```
                                   FCEE  31 008D5 163$:   BRW     101$                             ; 1693
            01    A6                 0C  8A 008D8 164$:   BICB2   #12, 1(R6)                       ; 1641
                  01         63      AB  91 008DC         CMPB    99(CCB), #1                      ; 1643
                                     06  12 008E0         BNEQ    165$                             ;
            01    A6                 04  88 008E2         BISB2   #4, 1(R6)                        ; 1645
                                     57  11 008E6         BRB     175$                             ;
    06            66                 04  E1 008E8 165$:   BBC     #4, (R6), 166$                   ; 1648
    4F            69                 0B  E0 008EC         BBS     #11, (R9), 175$                  ;
                                     48  11 008F0         BRB     174$                             ; 1650
    49            69                 0B  E0 008F2 166$:   BBS     #11, (R9), 175$                  ; 1651
    45            66                 09  E1 008F6         BBC     #9, (R6), 175$                   ;
    41            66                 04  E0 008FA         BBS     #4, (R6), 175$                   ; 1652
                  02         63      AB  91 008FE         CMPB    99(CCB), #2                      ;
                                     3B  12 00902         BNEQ    175$                             ;
            01    A6                 08  88 00904         BISB2   #8, 1(R6)                        ; 1654
                                     35  11 00908         BRB     175$                             ; 1635
                  01         63      AB  91 0090A 167$:   CMPB    99(CCB), #1                      ; 1660
                                     28  11 0090E         BRB     173$                             ;
                  02         63      AB  91 00910 168$:   CMPB    99(CCB), #2                      ; 1664
                                     29  13 00914         BEQL    175$                             ;
                  03         63      AB  91 00916         CMPB    99(CCB), #3                      ;
                                     1C  11 0091A         BRB     173$                             ;
                  02         63      AB  91 0091C 169$:   CMPB    99(CCB), #2                      ; 1670
                                     18  12 00920         BNEQ    174$                             ;
    19            69                 0B  E1 00922         BBC     #11, (R9), 175$                  ;
                                     12  11 00926         BRB     174$                             ; 1672
                  04         63      AB  91 00928 170$:   CMPB    99(CCB), #4                      ; 1676
                                     0A  11 0092C         BRB     173$                             ;
                  06         63      AB  91 0092E 171$:   CMPB    99(CCB), #6                      ; 1682
                                     04  11 00932         BRB     173$                             ;
                  05         63      AB  91 00934 172$:   CMPB    99(CCB), #5                      ; 1688
                                     05  13 00938 173$:   BEQL    175$                             ;
                                     2C  DD 0093A 174$:   PUSHL   #44                              ; 1690
                                   01E3  31 0093C         BRW     214$                             ;
                             E4      AB  D5 0093F 175$:   TSTL    -28(CCB)                         ; 1700
                                     07  12 00942         BNEQ    176$                             ;
            E4    AB         7C      AB  D0 00944         MOVL    124(CCB), -28(CCB)               ; 1702
                                     17  11 00949         BRB     178$                             ;
                             7C      AB  D5 0094B 176$:   TSTL    124(CCB)                         ; 1705
                                     12  13 0094E         BEQL    178$                             ;
                  50         E4      AB  D0 00950         MOVL    -28(CCB), R0                     ;
            7C    AB                 50  D1 00954         CMPL    R0, 124(CCB)                     ;
                                     04  15 00958         BLEQ    177$                             ;
                  50         7C      AB  D0 0095A         MOVL    124(CCB), R0                     ;
            E4    AB                 50  D0 0095E 177$:   MOVL    R0, -28(CCB)                     ;
    11    0087    CB                 04  E0 00962 178$:   BBS     #4, 135(CCB), 179$               ; 1724
    0B    0084    CB                 02  E0 00968         BBS     #2, 132(CCB), 179$               ; 1725
                             7A      AB  B5 0096E         TSTW    122(CCB)                         ; 1727
                                     06  12 00971         BNEQ    179$                             ;
            7A    AB       0080      CB  B0 00973         MOVW    128(CCB), 122(CCB)               ; 1729
                  10                 55  E8 00979 179$:   BLBS    V_DEFAULT_SIZE, 181$             ; 1731
    05            66                 0A  E0 0097C         BBS     #10, (R6), 180$                  ;
                  10                 64  91 00980         CMPB    (R4), #16                        ; 1732
                                     07  12 00983         BNEQ    181$                             ;
            7A    AB         D2      AB  B1 00985 180$:   CMPW    -46(CCB), 122(CCB)               ; 1735
                                     54  12 0098A         BNEQ    188$                             ;
                  52         D2      AB  9E 0098C 181$:   MOVAB   -46(CCB), R2                     ; 1740
```

```
          05      66            0A  E0 00990           BBS     #10, (R6), 182$                      : 1737
                  10            64  91 00994           CMPB    (R4), #16                            : 1738
                                06  12 00997           BNEQ    183$                                 :
                  62      7A    AB  B0 00999  182$:    MOVW    122(CCB), (R2)                       : 1740
                                1A  11 0099D           BRB     186$                                 :
                  50            62  3C 0099F  183$:    MOVZWL  (R2), R0                             : 1742
                  50      7A    AB  B1 009A2           CMPW    122(CCB), R0                         :
                                04  1B 009A6           BLEQU   184$                                 :
                  50      7A    AB  3C 009A8           MOVZWL  122(CCB), R0                         :
                  50      D2    AD  B1 009AC  184$:    CMPW    XAB_BLOCK+10, R0                     :
                                04  1B 009B0           BLEQU   185$                                 :
                  50      D2    AD  3C 009B2           MOVZWL  XAB_BLOCK+10, R0                     :
                  62            50  B0 009B6  185$:    MOVW    R0, (R2)                             :
                  20            64  91 009B9  186$:    CMPB    (R4), #32                            : 1744
                                29  12 009BC           BNEQ    190$                                 :
          25      66            0A  E0 009BE           BBS     #10, (R6), 190$                      :
                  53      7A    AB  3C 009C2           MOVZWL  122(CCB), R3                         : 1751
                                10  12 009C6           BNEQ    187$                                 :
                  1C            55  E9 009C8           BLBC    V_DEFAULT_SIZE, 190$                 : 1754
                  50    0082    CB  9A 009CB           MOVZBL  130(CCB), R0                         : 1756
          62      50    0200    8F  A5 009D0           MULW3   #512, R0, (R2)                       :
                                0F  11 009D6           BRB     190$                                 : 1751
                  09            55  E8 009D8  187$:    BLBS    V_DEFAULT_SIZE, 189$                 : 1764
                  53            62  B1 009DB           CMPW    (R2), R3                             : 1765
                                04  1B 009DE           BLEQU   189$                                 :
                                25  DD 009E0  188$:    PUSHL   #37                                  : 1767
                                63  11 009E2           BRB     198$                                 :
                  62            53  B0 009E4  189$:    MOVW    R3, (R2)                             : 1769
                  20            64  91 009E7  190$:    CMPB    (R4), #32                            : 1780
                                62  12 009EA           BNEQ    200$                                 :
                  50      18    AE  D0 009EC           MOVL    24(SP), KEY_DEFN                     : 1793
                                06  13 009F0           BEQL    191$                                 : 1795
                  53      02    A0  32 009F2           CVTWL   2(KEY_DEFN), KEY_COUNT               :
                                02  11 009F6           BRB     192$                                 :
                                53  D4 009F8  191$:    CLRL    KEY_COUNT                            :
                  52            01  D0 009FA  192$:    MOVL    #1, XAB_STATUS                       : 1799
                  58      CC    AD  D0 009FD           MOVL    XAB_BLOCK+4, KEY_XAB                 : 1800
                                58  D5 00A01  193$:    TSTL    KEY_XAB                              : 1802
                                3D  13 00A03           BEQL    197$                                 :
                                53  D5 00A05           TSTL    KEY_COUNT                            :
                                39  15 00A07           BLEQ    197$                                 :
                  15            68  91 00A09           CMPB    (KEY_XAB), #21                       : 1805
                                F3  12 00A0C           BNEQ    193$                                 :
        4E        A8      1E    A8  B1 00A0E           CMPW    30(KEY_XAB), 78(KEY_XAB)             : 1809
                                07  12 00A13           BNEQ    194$                                 :
        4D        A8      2E    A8  91 00A15           CMPB    46(KEY_XAB), 77(KEY_XAB)             : 1810
                                03  13 00A1A           BEQL    195$                                 :
                  52            31  D0 00A1C  194$:    MOVL    #49, XAB_STATUS                      : 1812
          13      A8      4C    A8  91 00A1F  195$:    CMPB    76(KEY_XAB), 19(KEY_XAB)             : 1814
                                03  13 00A24           BEQL    196$                                 :
                  52            31  D0 00A26           MOVL    #49, XAB_STATUS                      : 1816
                  54      04    A8  D0 00A29  196$:    MOVL    4(KEY_XAB), NEXT                     : 1823
                                58  DD 00A2D           PUSHL   KEY_XAB                              : 1824
                  7E      50    8F  9A 00A2F           MOVZBL  #80, -(SP)                           :
   00000000G      00            02  FB 00A33           CALLS   #2, FOR$$FREE_VM                     :
                  58            54  D0 00A3A           MOVL    NEXT, KEY_XAB                        : 1825
                  53            03  C2 00A3D           SUBL2   #3, KEY_COUNT                        : 1827
```

```
                                         BF   11  00A40          BRB      193$                              ; 1802
                     09                  52   E8  00A42  197$:   BLBS     XAB_STATUS, 200$                  ; 1839
                                         52   DD  00A45          PUSHL    XAB_STATUS                        ; 1841
                                       00D8   31  00A47  198$:   BRW      214$                              ;
                     01   A6             20   8A  00A4A  199$:   BICB2    #32, 1(R6)                        ; 1856
                          28             55   E9  00A4E  200$:   BLBC     V_DEFAULT_SIZE, 202$              ; 1864
          22   0087       CB             04   E0  00A51          BBS      #4, 135(CCB), 202$                ; 1866
          1C   0084       CB             02   E0  00A57          BBS      #2, 132(CCB), 202$                ; 1867
                                0080     CB   B5  00A5D          TSTW     128(CCB)                          ; 1868
                                         16   13  00A61          BEQL     202$                              ;
                          50    0080     CB   B0  00A63          MOVW     128(CCB), NEW_RECL                ; 1873
          03              66             0B   E1  00A68          BBC      #11, (R6), 201$                   ; 1874
                          50             04   A2  00A6C          SUBW2    #4, NEW_RECL                      ; 1876
                     D2   AB             50   B1  00A6F  201$:   CMPW     NEW_RECL, -46(CCB)                ; 1877
                                         04   1E  00A73          BGEQU    202$                              ;
                     D2   AB             50   B0  00A75          MOVW     NEW_RECL, -46(CCB)                ; 1879
                          05    00CA     CB   E9  00A79  202$:   BLBC     202(CCB), 203$                    ; 1888
                     00   BE      28     AE   90  00A7E          MOVB     ORIG_RAT, @0(SP)                  ; 1890
                          06             55   E8  00A83  203$:   BLBS     V_DEFAULT_SIZE, 204$              ; 1897
                          50      D2     AB   3C  00A86          MOVZWL   -46(CCB), R0                      ;
                                         0E   11  00A8A          BRB      206$                              ;
                          06      00     BE   E9  00A8C  204$:   BLBC     @0(SP), 205$                      ; 1898
                          50      51     8F   9A  00A90          MOVZBL   #81, R0                           ;
                                         04   11  00A94          BRB      206$                              ;
                          50      50     8F   9A  00A96  205$:   MOVZBL   #80, R0                           ;
                     D4   AB             50   B0  00A9A  206$:   MOVW     R0, -44(CCB)                      ; 1897
                          04      00     BE   E9  00A9E          BLBC     @0(SP), 207$                      ; 1905
                          69      80     8F   88  00AA2          BISB2    #128, (R9)                        ; 1907
          04         00   BE             01   E1  00AA6  207$:   BBC      #1, @0(SP), 208$                  ; 1908
                          69      40     8F   88  00AAB          BISB2    #64, (R9)                         ; 1910
          04         00   BE             02   E1  00AAF  208$:   BBC      #2, @0(SP), 209$                  ; 1911
                     01   A9             01   88  00AB4          BISB2    #1, 1(R9)                         ; 1913
                          7E      D2     AB   3C  00AB8  209$:   MOVZWL   -46(CCB), -(SP)                   ; 1920
               00000000G  00             01   FB  00ABC          CALLS    #1, FOR$$GET_VM                   ;
                     EC   AB             50   D0  00AC3          MOVL     R0, -20(CCB)                      ;
                          7E      1C     BE   9A  00AC7          MOVZBL   @28(SP), -(SP)                    ; 1935
               00000000G  00             01   FB  00ACB          CALLS    #1, FOR$$GET_VM                   ;
                          57             50   D0  00AD2          MOVL     R0, T                             ;
                          50      1C     BE   9A  00AD5          MOVZBL   @28(SP), R0                       ; 1936
                          58      14     BE   D0  00AD9          MOVL     @20(SP), R8                       ;
          67              68             50   28  00ADD          MOVC3    R0, (R8), (T)                     ;
                     14   BE             57   D0  00AE1          MOVL     T, @20(SP)                        ; 1937
                     24   BE             57   D0  00AE5          MOVL     T, @36(SP)                        ; 1938
                     20   BE             57   D0  00AE9          MOVL     T, @32(SP)                        ; 1939
                   009F   CB      1C     BE   90  00AED          MOVB     @28(SP), 159(CCB)                 ; 1940
                     FE   AB             01   88  00AF3          BISB2    #1, -2(CCB)                       ; 1941
                          50      61     AB   9A  00AF7          MOVZBL   97(CCB), R0                       ; 1948
                                         06   12  00AFB          BNEQ     210$                              ; 1951
                     C4   AB             01   90  00AFD          MOVB     #1, -60(CCB)                      ; 1952
                                         27   11  00B01          BRB      215$                              ;
                          10             50   91  00B03  210$:   CMPB     R0, #16                           ; 1954
                                         06   12  00B06          BNEQ     211$                              ;
                     C4   AB             02   90  00B08          MOVB     #2, -60(CCB)                      ; 1955
                                         1C   11  00B0C          BRB      215$                              ;
                          20             50   91  00B0E  211$:   CMPB     R0, #32                           ; 1957
                                         0D   12  00B11          BNEQ     213$                              ;
          03              66             0B   E1  00B13          BBC      #11, (R6), 212$                   ; 1960
```

```
                                       FE20  31 00B17          BRW     174$                                  ;
                     C4   AB             03  90 00B1A 212$:    MOVB    #3, -60(CCB)                          ; 1962
                                         0A  11 00B1E          BRB     215$                                  ; 1948
                                         33  DD 00B20 213$:    PUSHL   #51                                   ; 1966
              00000000G   00             01  FB 00B22 214$:    CALLS   #1, FOR$$SIGNAL_STO                   ;
                                         04  00B29             RET                                           ;
                     24   AB     EC  AB  D0 00B2A 215$:        MOVL    -20(CCB), 36(CCB)                     ; 1973
                     20   AB     D2  AB  B0 00B2F              MOVW    -46(CCB), 32(CCB)                     ; 1974
                     9C   AB     EC  AB  D0 00B34              MOVL    -20(CCB), -100(CCB)                   ; 1975
         4C               69         0B  E0 00B39              BBS     #11, (R9), 217$                       ; 1982
         48               66         04  E0 00B3D              BBS     #4, (R6), 217$                        ; 1983
         44               66         0A  E0 00B41              BBS     #10, (R6), 217$                       ; 1984
                          3F   00CA  CB  E8 00B45              BLBS    202(CCB), 217$                        ; 1985
         3A          0C   BE         06  E0 00B4A              BBS     #6, @12(SP), 217$                     ; 1986
         34        0087   CB         04  E1 00B4F              BBC     #4, 135(CCB), 217$                    ; 1991
                          7E   0190  8F  3C 00B55              MOVZWL  #400, -(SP)                           ; 2003
              00000000G   00         01  FB 00B5A              CALLS   #1, FOR$$GET_VM                       ;
                     C8   AB         50  D0 00B61              MOVL    RCE, -56(CCB)                         ; 2010
                     CC   AB         50  D0 00B65              MOVL    RCE, -52(CCB)                         ; 2011
                          51   017C  C0  9E 00B69              MOVAB   380(R0), OLD_RCE                      ; 2012
                          52         14  D0 00B6E              MOVL    #20, I                                ; 2013
                          61         50  D0 00B71 216$:        MOVL    RCE, (OLD_RCE)                        ; 2015
                     04   A0         51  D0 00B74              MOVL    OLD_RCE, 4(RCE)                       ; 2016
                                 08  A0  D4 00B78              CLRL    8(RCE)                                ; 2017
                          51         80  7E 00B7B              MOVAQ   (RCE)+, OLD_RCE                       ; 2018
                          50         0C  C0 00B7E              ADDL2   #12, RCE                              ; 2019
                                     52  D7 00B81              DECL    I                                     ; 2013
                                     EC  12 00B83              BNEQ    216$                                  ;
                     01   A9         20  88 00B85              BISB2   #32, 1(R9)                            ; 2022
                     D8   AB         02  90 00B89 217$:        MOVB    #2, -40(CCB)                          ; 2029
                          66         01  88 00B8D              BISB2   #1, (R6)                              ; 2030
                          07 00000000G   00  E8 00B90          BLBS    FOR$$L_XIT_LOCK, 218$                 ; 2036
              00000000G   00         00  FB 00B97              CALLS   #0, FOR$$DECL_EXITH                   ;
                                     04  00B9E 218$:           RET                                           ; 2039

; Routine Size:  2975 bytes,    Routine Base:  _FOR$CODE + 0090


; 1999          2040  1
; 2000          2041  1 END                                        ! End of FOR$$OPEN_DEFLT module
; 2001          2042  1
; 2002          2043  0 ELUDOM




;                               PSECT SUMMARY
;
;       Name                   Bytes                      Attributes
;
;   _FOR$CODE                     3119  NOVEC,NOWRT,  RD ,  EXE,  SHR,  LCL,  REL,  CON,  PIC,ALIGN(2)
;                                   Library Statistics
;
;                                            -------- Symbols --------       Pages        Processing
;          File                              Total      Loaded   Percent      Mapped       Time
;
;    _$255$DUA28:[SYSLIB]STARLET.L32;1        9776        127        1         581         00:01.1
;    _$255$DUA28:[FORRTL.OBJ]FORLIB.L32;1      711        283       39          52         00:00.6
;    _$255$DUA28:[FORRTL.OBJ]RTLLIB.L32;1       36          0        0           8         00:00.1



;                                           COMMAND QUALIFIERS

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

; Size:          3012 code + 107 data bytes
; Run Time:         01:26.6
; Elapsed Time:     03:27.2
; Lines/CPU Min:     1414
; Lexemes/CPU-Min: 15974
; Memory Used:  1167 pages
; Compilation Complete
```

0182 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

FORRAB
LIS

FORREADDF
LIS

FOROPNKEY
LIS

FORPAUSE
LIS

FORRANDOM
LIS

FOROPEN
LIS

FOROPENDE
LIS

FORREADDO
LIS