```
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**COMFDBSET                              L 14

  CCCCCCCC    OOOOOO    MM      MM  FFFFFFFFFF  DDDDDDDD    BBBBBBBB      SSSSSSSS  EEEEEEEEEE  TTTTTTTTTT
  CCCCCCCC    OOOOOO    MM      MM  FFFFFFFFFF  DDDDDDDD    BBBBBBBB      SSSSSSSS  EEEEEEEEEE  TTTTTTTTTT
CC          OO      OO  MMMM  MMMM  FF          DD      DD  BB      BB  SS          EE              TT
CC          OO      OO  MMMM  MMMM  FF          DD      DD  BB      BB  SS          EE              TT
CC          OO      OO  MM  MM  MM  FF          DD      DD  BB      BB  SS          EE              TT
CC          OO      OO  MM  MM  MM  FF          DD      DD  BB      BB  SS          EE              TT
CC          OO      OO  MM      MM  FFFFFFFF    DD      DD  BBBBBBBB      SSSSSS    EEEEEEEE        TT
CC          OO      OO  MM      MM  FFFFFFFF    DD      DD  BBBBBBBB      SSSSSS    EEEEEEEE        TT
CC          OO      OO  MM      MM  FF          DD      DD  BB      BB          SS  EE              TT
CC          OO      OO  MM      MM  FF          DD      DD  BB      BB          SS  EE              TT
CC          OO      OO  MM      MM  FF          DD      DD  BB      BB          SS  EE              TT          ....
CC          OO      OO  MM      MM  FF          DD      DD  BB      BB          SS  EE              TT          ....
  CCCCCCCC    OOOOOO    MM      MM  FF          DDDDDDDD    BBBBBBBB    SSSSSSSS    EEEEEEEEEE      TT          ....
  CCCCCCCC    OOOOOO    MM      MM  FF          DDDDDDDD    BBBBBBBB    SSSSSSSS    EEEEEEEEEE      TT          ....

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
```

```
;     1          0001  0 MODULE COM$FDBSET (%TITLE'Compatibility CALL FDBSET'
;     2          0002  0         IDENT = '1-006',            ! File: COMFDBSET.B32  Edit: SBL1006
;     3          0003  0         LINKAGE (FORTRAN)           ! Call-by-reference
;     4          0004  0         ) =
;     5          0005  1 BEGIN
;     6          0006  1
;     7          0007  1 !
;     8          0008  1 !****************************************************************************
;     9          0009  1 !*                                                                          *
;    10          0010  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
;    11          0011  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
;    12          0012  1 !*  ALL RIGHTS RESERVED.                                                    *
;    13          0013  1 !*                                                                          *
;    14          0014  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
;    15          0015  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
;    16          0016  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
;    17          0017  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
;    18          0018  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
;    19          0019  1 !*  TRANSFERRED.                                                            *
;    20          0020  1 !*                                                                          *
;    21          0021  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
;    22          0022  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
;    23          0023  1 !*  CORPORATION.                                                            *
;    24          0024  1 !*                                                                          *
;    25          0025  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
;    26          0026  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
;    27          0027  1 !*                                                                          *
;    28          0028  1 !*                                                                          *
;    29          0029  1 !****************************************************************************
;    30          0030  1 !
;    31          0031  1
;    32          0032  1 !++
;    33          0033  1 ! FACILITY:  FORTRAN Compatibility Library
;    34          0034  1 !
;    35          0035  1 ! ABSTRACT:
;    36          0036  1 !
;    37          0037  1 !       Set RMS FAB quantities which are equivalent
;    38          0038  1 !       of the FCS-11 FDB (File data block).
;    39          0039  1 !
;    40          0040  1 ! ENVIRONMENT:  User Mode - AST re-entrant
;    41          0041  1 !
;    42          0042  1 ! AUTHOR:  Thomas N. Hastings, CREATION DATE:  5-Dec-1977
;    43          0043  1 !
;    44          0044  1 ! MODIFIED BY:
;    45          0045  1 !
;    46          0046  1 !       Thomas N. Hastings, 4-Dec-1977: VERSION 0
;    47          0047  1 ! [Previous edit history removed.  SBL 1-Mar-1983]
;    48          0048  1 ! 1-001 - Update version number and copyright notice.  JBS 16-NOV-78
;    49          0049  1 ! 1-002 - Declare NULLPARAMETER for new BLISS compiler.  JBS 22-NOV-78
;    50          0050  1 ! 1-003 - Change REQUIRE file names from FOR... to OTS...  JBS 07-DEC-78
;    51          0051  1 ! 1-004 - Change OPEN prefix to LUB.  JBS 13-DEC-78
;    52          0052  1 ! 1-005 - Set FAB$V_UPI when SHARED.  SPR 11-27878  SBL 27-Dec-1979
;    53          0053  1 ! 1-006 - Use prologue file.  SBL 1-Mar-1983
;    54          0054  1 !--
;    56      0055  1 !
;    57      0056  1 ! PROLOGUE FILE:
;    58      0057  1 !
;    59      0058  1
;    60      0059  1 REQUIRE 'RTLIN:FORPROLOG';              ! FOR$ declarations
;    61      0125  1
;    62      0126  1 !
;    63      0127  1 ! TABLE OF CONTENTS:
;    64      0128  1 !
;    65      0129  1
;    66      0130  1 FORWARD ROUTINE
;    67      0131  1         FDBSET: NOVALUE;                ! Set FDB equivalent data
;    68      0132  1
;    69      0133  1 !
;    70      0134  1 ! MACROS:
;    71      0135  1 !
;    72      0136  1
;    73      0137  1 MACRO
;    74      0138  1         BYTE_BY_REF = 0, 0, 8, 0 %;     ! First byte in descriptor.
;    75      0139  1                                         ! used to fetch first byte of character by-reference.
;    76      0140  1 !
;    77      0141  1 ! EQUATED SYMBOLS:
;    78      0142  1 !
;    79      0143  1
;    80      0144  1 !
;    81      0145  1 ! OWN STORAGE:
;    82      0146  1 !
;    83      0147  1
;    84      0148  1 !
;    85      0149  1 ! EXTERNAL REFERENCES:
;    86      0150  1 !
;    87      0151  1
;    88      0152  1 EXTERNAL ROUTINE
;    89      0153  1         FOR$$ERR_OPECLO: NOVALUE ADDRESSING_MODE (GENERAL),     ! Error handler for OPEN/CLOSE
;    90      0154  1         FOR$$CB_PUSH: JSB_CB_PUSH NOVALUE ADDRESSING_MODE (GENERAL),    ! Allocate LUB/ISB/RAB if necessary
;    91      0155  1         FOR$$CB_POP: JSB_CB_POP NOVALUE ADDRESSING_MODE (GENERAL),      ! Pop LUB/ISB/RAB
;    92      0156  1         FOR$$GET_VM: ADDRESSING_MODE (GENERAL),                 ! Allocate virtual memory
;    93      0157  1         FOR$$SIGNAL_STO: NOVALUE ADDRESSING_MODE (GENERAL);     ! SIGNAL_STOP errors
;   95      0158  1 GLOBAL ROUTINE FDBSET ( ! FORTRAN compatibility set FDB equivalent
;   96      0159  1         LOGICAL_UNIT,    ! Adr. of logical unit no.
;   97      0160  1         MODE_LITERAL,    ! Adr. of descr. of mode literal string or adr. of character
;   98      0161  1         SHARE_LITERAL,   ! Adr. of descr. of share literal string or adr. of character
;   99      0162  1         NUM_BUF,         ! Adr. of number of buffers
;  100      0163  1         INIT_ALLOC,      ! Adr. of initial block to allocate
;  101      0164  1         EXTEND_BLK_CNT)  ! Adr. of number of blocks to extend
;  102      0165  1         : NOVALUE =      ! No value is returned
;  103      0166  1
;  104      0167  1 !++
;  105      0168  1 ! FUNCTIONAL DESCRIPTION:
;  106      0169  1 !
;  107      0170  1 !       Set RMS FAB (File access block) and Language support library LUB
;  108      0171  1 !       (logical unit block) with OPFN-like parameters for unit LOGICAL_UNIT.
;  109      0172  1 !       This procedure is provided for compatibility with CALL FDBSET
;  110      0173  1 !       on PDP-11 systems with FCS-11 which set FDB (File descriptor block).
;  111      0174  1
;  112      0175  1 ! FORMAL PARAMETERS:
;  113      0176  1 !
;  114      0177  1 !       LOGICAL_UNIT.rw.r       Adr. of word containing logical unit number.
;  115      0178  1 !       [MODE_LITERAL.rt.{ds:r} Adr. of string descr. which can be:
;  116      0179  1 !                               'NEW', 'OLD', 'READONLY', 'APPEND', 'MODIFY',
;  117      0180  1 !                               'INH.BIT SUPERSEDE', 'UNKNOWN'
;  118      0181  1 !                               Only the first char is looked at.
;  119      0182  1 !       [SHARE_LITERAL.rt.{ds:r}        Adr. of string descr. which can be: 'SHARE'
;  120      0183  1 !                               Only the first character is looked at.
;  121      0184  1 !       [NUM_BUF.rw.r           Adr. of word containing number of buffers
;  122      0185  1 !       [INIT_ALLOC.rw.r        Adr. of word containing no of
;  123      0186  1 !                               blocks for initial allocation
;  124      0187  1 !       [EXTEND_BLK_CNT.rw.r]]]]]       Adr. of word containing no. of blocks
;  125      0188  1 !                               for file extended allocation.
;  126      0189  1 !
;  127      0190  1 ! IMPLICIT INPUTS:
;  128      0191  1 !
;  129      0192  1 !       OTS$$A_CUR_LUB          Contains address of current LUB/ISB/RAB or 0.
;  130      0193  1 !       OTS$$AA_LUB_TAB Table of LUB/ISB/RABs by logical unit
;  131      0194  1 !
;  132      0195  1 ! IMPLICIT OUTPUTS:
;  133      0196  1 !
;  134      0197  1 !       FAB allocated and set, LUB/ISB/RAB allocated and LUB set
;  135      0198  1 !
;  136      0199  1 ! ROUTINE VALUE:
;  137      0200  1 ! COMPLETION CODES:
;  138      0201  1 !
;  139      0202  1 !       NONE
;  140      0203  1 !
;  141      0204  1 ! SIDE EFFECTS:
;  142      0205  1 !
;  143      0206  1 !       Allocates LUB/ISB/RAB and/or FAB, if not already allocated.
;  144      0207  1 !--
;  145      0208  1
;  146      0209  2     BEGIN
;  147      0210  2
;  148      0211  2     BUILTIN NULLPARAMETER;
;  149      0212  2
;  150      0213  2     GLOBAL REGISTER
;  151      0214  2         CCB = 11: REF $FOR$CCB_DECL;
;  152        0215  2
;  153        0216  2         MAP
;  154        0217  2             LOGICAL_UNIT: REF VECTOR[1, WORD],
;  155        0218  2             MODE_LITERAL: REF DSC$DESCRIPTOR,
;  156        0219  2             SHARE_LITERAL: REF DSC$DESCRIPTOR,
;  157        0220  2             NUM_BUF: REF VECTOR[1, WORD],
;  158        0221  2             INIT_ALLOC: REF VECTOR[1, WORD],
;  159        0222  2             EXTEND_BLK_CNT: REF VECTOR[1,WORD];
;  160        0223  2         LOCAL
;  161        0224  2             FAB: REF BLOCK [FAB$C_BLN, BYTE],        ! base pointer to FAB
;  162        0225  2             L_UNWIND_ACTION: VOLATILE;               ! Action flag for handler on unwind
;  163        0226  2         ENABLE
;  164        0227  2             FOR$$ERR_OPECLO (L_UNWIND_ACTION);       ! Don't pass 2nd arg (OPEN-CLOSE array)
;  165        0228  2                                                      ! since never an ERR=
;  166        0229  2
;  167        0230  2         !+
;  168        0231  2         ! Push down current logical unit block if any, and allocate LUB/ISB/RAB
;  169        0232  2         ! for this unit if not already allocated.  Unwind action is a no-op
;  170        0233  2         ! until LUB/ISB/RAB pushed down and LOGICAL_UNIT LUB flagged as current.
;  171        0234  2         ! Then unwind action on errors is to POP the LUB/ISB/RAB.
;  172        0235  2         ! On return CCB points to current control block.
;  173        0236  2         !-
;  174        0237  2
;  175        0238  2         L_UNWIND_ACTION = FOR$K_UNWINDNOP;
;  176        0239  2         FOR$$CB_PUSH (.LOGICAL_UNIT[0], LUB$K_LUN_MIN);
;  177        0240  2         L_UNWIND_ACTION = FOR$K_UNWINDPOP;
;  178        0241  2
;  179        0242  2         !+
;  180        0243  2         ! If unit already opened, SIGNAL_STOP FOR$_UNIALROPE (34='UNIT ALREADY OPEN')
;  181        0244  2         !-
;  182        0245  2
;  183        0246  2         IF .CCB[LUB$V_OPENED] THEN $FOR$$SIGNAL_STO (FOR$K_UNIALROPE);
;  184        0247  2
;  185        0248  2         !+
;  186        0249  2         ! Allocate zeroed FAB if not already allocated and save address in LUB.
;  187        0250  2         !-
;  188        0251  2
;  189        0252  2         IF (FAB = .CCB[LUB$A_FAB]) EQLA 0
;  190        0253  2         THEN
;  191        0254  3             BEGIN
;  192        0255  3             FAB = FOR$$GET_VM (FAB$K_BLN);
;  193        0256  3             CH$FILL (0, FAB$K_BLN, .FAB);
;  194        0257  3             FAB[FAB$B_BID] = FAB$C_BID;
;  195        0258  3             FAB[FAB$B_BLN] = FAB$K_BLN;
;  196        0259  3             CCB[LUB$A_FAB] = .FAB;
;  197        0260  2             END;
;  198        0261  2
;  199        0262  2         !+
;  200        0263  2         ! If MODE_LITERAL is present, dispatch on first character of string literal
;  201        0264  2         ! Accept either character by-descriptor or string by-descriptor.
;  202        0265  2         ! If first byte of descriptor is GEQ ASCII A, assume character by-reference
;  203        0266  2         ! otherwise string by-descriptor.
;  204        0267  2         !-
;  205        0268  2
;  206        0269  2         IF NOT NULLPARAMETER (2)
;  207        0270  2         THEN
;  208        0271  3             BEGIN
;  209        0272  3           BIND LEGAL_CHAR = UPLIT ('AIMNORU');
;  210        0273  3           LOCAL P;                    ! Temporary char pointer
;  211        0274  4           P = CH$FIND_CH (7, LEGAL_CHAR, (IF .MODE_LITERAL[BYTE_BY_REF] GEQ %C'A'
;  212        0275  4                                       THEN
;  213        0276  4                                           .MODE_LITERAL[BYTE_BY_REF]                  ! character by-ref
;  214        0277  4                                       ELSE
;  215        0278  3                                           .(.MODE_LITERAL[DSC$A_POINTER])));  ! character by-descr.
;  216        0279  3           IF CH$FAIL (.P) THEN $FOR$$SIGNAL_STO (FOR$K_INVARGFOR);
;  217        0280  3
;  218        0281  3           CASE .P - LEGAL_CHAR FROM 0 TO 6 OF
;  219        0282  3               SET
;  220        0283  3               [0]:                            ! 'APPEND'
;  221        0284  4                   BEGIN
;  222        0285  4                   CCB[RAB$V_EOF] = 1;
;  223        0286  4                   CCB[RAB$B_RAC] = RAB$C_SEQ;
;  224        0287  4                   CCB[LUB$V_OLD_FILE] = 1;
;  225        0288  4                   CCB[LUB$V_APPEND] = 1;
;  226        0289  3                   END;
;  227        0290  3
;  228        0291  3               [1]:                            ! \\\'INHIBIT SUPERSEDE\\\
;  229        0292  3                           ;                   ! Nothing for now\\\
;  230        0293  3
;  231        0294  3               [2]:                            ! 'MODIFY'
;  232        0295  3                   CCB[LUB$V_OLD_FILE] = 1;
;  233        0296  3
;  234        0297  3               [3]:                            ! 'NEW'
;  235        0298  3                           ;                   ! nothing special to do
;  236        0299  3
;  237        0300  3               [4]:                            ! 'OLD'
;  238        0301  3                   CCB[LUB$V_OLD_FILE] = 1;
;  239        0302  3
;  240        0303  3               [5]:                            ! 'READONLY'
;  241        0304  4                   BEGIN
;  242        0305  4                   FAB[FAB$B_FAC] = FAB$M_GET;
;  243        0306  4                   CCB[LUB$V_OLD_FILE] = 1;
;  244        0307  4                   CCB[LUB$V_READ_ONLY] = 1;
;  245        0308  3                   END;
;  246        0309  3
;  247        0310  3               [6]:                            ! 'UNKNOWN'
;  248        0311  3                   FAB[FAB$V_CIF] = 1;
;  249        0312  3
;  250        0313  3               TES;
;  251        0314  2           END;                            ! End MODE_LITERAL
;  252        0315  2
;  253        0316  2       !+
;  254        0317  2       ! SHARE_LITERAL.  If present and first character equal to 'S',
;  255        0318  2       ! set for sharing put, get, and update.
;  256        0319  2       ! If MODE_LITERAL is present, dispatch on first character of string literal
;  257        0320  2       ! Accept either character by-descriptor or string by-descriptor.
;  258        0321  2       ! If first byte of descriptor is GEQ ASCII A, assume character by-reference
;  259        0322  2       ! otherwise string by-descriptor.
;  260        0323  2       !-
;  261        0324  2
;  262        0325  2       IF NOT NULLPARAMETER (3)
;  263        0326  2       THEN
;  264        0327  3           BEGIN
;  265        0328  4           IF (IF .SHARE_LITERAL[BYTE_BY_REF] GEQ %C'A'
;  266           0329  4                  THEN
;  267           0330  4                      .SHARE_LITERAL[BYTE_BY_REF]                          ! character by-reference
;  268           0331  4                  ELSE
;  269           0332  4                      .(.SHARE_LITERAL[DSC$A_POINTER])<0,8>)               ! string by-descr.
;  270           0333  3                  NEQ %C'S' THEN $FOR$$SIGNAL_STO (FOR$K_INVARGFOR);
;  271           0334  3
;  272           0335  3              FAB[FAB$B_SHR] = FAB$M_GET + FAB$M_PUT + FAB$M_UPD + FAB$M_UPI;
;  273           0336  2              END;
;  274           0337  2
;  275           0338  2          !+
;  276           0339  2          ! NUM_BUF.  If number of buffers arg is present, set RAB buffer count.
;  277           0340  2          !-
;  278           0341  2
;  279           0342  2          IF NOT NULLPARAMETER (4) THEN CCB[RAB$B_MBF] = .NUM_BUF[0];
;  280           0343  2
;  281           0344  2          !+
;  282           0345  2          ! INIT_ALLOC.  If initial allocation arg is present, set FAB$L_ALQ
;  283           0346  2          ! with magnitude of arg and set contiguous best try bit.
;  284           0347  2          !-
;  285           0348  2
;  286           0349  2          IF NOT NULLPARAMETER (5)
;  287           0350  2          THEN
;  288           0351  3              BEGIN
;  289           0352  3              FAB[FAB$L_ALQ] = ABS (.INIT_ALLOC[0]);
;  290           0353  3              FAB[FAB$V_CBT] = 1;
;  291           0354  2              END;
;  292           0355  2
;  293           0356  2          !+
;  294           0357  2          ! EXTEND_BLK_CNT.  If extend allocation count is present, set FAB$W_DEQ.
;  295           0358  2          !-
;  296           0359  2
;  297           0360  2          IF NOT NULLPARAMETER (6) THEN FAB[FAB$W_DEQ] = ABS (.EXTEND_BLK_CNT[0]);
;  298           0361  2
;  299           0362  2          !+
;  300           0363  2          ! Pop (but do not deallocate) LUB/ISB/RAB and return with FAB allocated.
;  301           0364  2          !-
;  302           0365  2
;  303           0366  2          FOR$$CB_POP ();
;  304           0367  2          RETURN;
;  305           0368  1          END;


                                                                   .TITLE  COM$FDBSET Compatibility CALL FDBSET
                                                                   .IDENT  \1-006\

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

                         00  55  52  4F  4E  4D  49  41  00000 P.AAA:  .ASCII  \AIMNORU\<0>                                  ;

                                                            LEGAL_CHAR=         P.AAA
                                                                   .EXTRN  FOR$$ERR_OPECLO
                                                                   .EXTRN  FOR$$CB_PUSH, FOR$$CB_POP
                                                                   .EXTRN  FOR$$GET_VM, FOR$$SIGNAL_STO

                                         08FC 00000                .ENTRY  FDBSET, Save R2,R3,R4,R5,R6,R7,R11            ; 0158
                                       7E  D4 00002                CLRL    L_UNWIND_ACTION                               ; 0209
                         6D       0119  CF  DE 00004                MOVAL   2T$, (FP)                                     ;
```

```
                                         6E            01 D0 00009         MOVL    #1, L_UNWIND_ACTION                         ; 0238
                                                       50 D4 0000C         CLRL    R0                                          ; 0239
                                         52       04   BC 3C 0000E         MOVZWL  @LOGICAL_UNIT, R2                           ;
                                           00000000G   00 16 00012         JSB     FOR$$CB_PUSH                                ;
                                                       6E D4 00018         CLRL    L_UNWIND_ACTION                             ; 0240
                                         57       FC   AB 9E 0001A         MOVAB   -4(CCB), R7                                 ; 0246
                                         05            67 E9 0001E         BLBC    (R7), 1$                                    ;
                                                       22 DD 00021         PUSHL   #34                                         ;
                                                     00B0 31 00023         BRW     15$                                         ;
                                         56       E8   AB D0 00026 1$:     MOVL    -24(CCB), FAB                               ; 0252
                                                       1F 12 0002A         BNEQ    2$                                          ;
                                         7E       50   8F 9A 0002C         MOVZBL  #80, -(SP)                                  ; 0255
                             00000000G   00            01 FB 00030         CALLS   #1, FOR$$GET_VM                             ;
                                         56            50 D0 00037         MOVL    R0, FAB                                     ;
      0050   8F            00            6E            00 2C 0003A         MOVC5   #0, (SP), #0, #80, (FAB)                    ; 0256
                                                       66    00041                                                             ;
                                         66     5003   8F B0 00042         MOVW    #20483, (FAB)                               ; 0257
                                    E8   AB            56 D0 00047         MOVL    FAB, -24(CCB)                               ; 0259
                                         02            6C 91 0004B 2$:     CMPB    (AP), #2                                    ; 0269
                                                       61 1F 0004E         BLSSU   11$                                         ;
                                                  08   AC D5 00050         TSTL    8(AP)                                       ;
                                                       5C 13 00053         BEQL    11$                                         ;
                                         50       08   AC D0 00055         MOVL    MODE_LITERAL, R0                            ; 0274
                                    41   8F            60 91 00059         CMPB    (R0), #65                                   ;
                                                       05 1F 0005D         BLSSU   3$                                          ;
                                         50            60 9A 0005F         MOVZBL  (R0), R0                                    ; 0276
                                                       04 11 00062         BRB     4$                                          ;
                                         50       04   B0 D0 00064 3$:     MOVL    @4(R0), R0                                  ; 0278
                      8B   AF            07            50 3A 00068 4$:     LOCC    R0, #7, LEGAL_CHAR                          ; 0274
                                                       02 12 0006D         BNEQ    5$                                          ;
                                                       51 D4 0006F         CLRL    R1                                          ;
                                         52            51 D0 00071 5$:     MOVL    R1, P                                       ;
                                                       5E 13 00074         BEQL    14$                                         ; 0279
                                         50     FF7E   CF 9E 00076         MOVAB   LEGAL_CHAR, R0                              ; 0281
                           51            52            50 C3 0007B         SUBL3   R0, P, R1                                   ;
                           06            00            51 CF 0007F         CASEL   R1, #0, #6                                  ;
           002E          001C          002E          000E    00083 6$:     .WORD   7$-6$,-                                     ;
                         002A          0021          001C    0008B                 11$-6$,-                                    ;
                                                                                   8$-6$,-                                     ;
                                                                                   11$-6$,-                                    ;
                                                                                   8$-6$,-                                     ;
                                                                                   9$-6$,-                                     ;
                                                                                   10$-6$                                      ;
                                    05   AB            01 88 00091 7$:     BISB2   #1, 5(CCB)                                  ; 0285
                                                  1E   AB 94 00095         CLRB    30(CCB)                                     ; 0286
                                         67     2008   8F A8 00098         BISW2   #8200, (R7)                                 ; 0288
                                                       12 11 0009D         BRB     11$                                         ; 0281
                                         67            08 88 0009F 8$:     BISB2   #8, (R7)                                    ; 0301
                                                       0D 11 000A2         BRB     11$                                         ;
                                    16   A6            02 90 000A4 9$:     MOVB    #2, 22(FAB)                                 ; 0305
                                         67            0C 88 000A8         BISB2   #12, (R7)                                   ; 0307
                                                       04 11 000AB         BRB     11$                                         ; 0281
                                    07   A6            02 88 000AD 10$:    BISB2   #2, 7(FAB)                                  ; 0311
                                         03            6C 91 000B1 11$:    CMPB    (AP), #3                                    ; 0325
                                                       2D 1F 000B4         BLSSU   17$                                         ;
                                                  0C   AC D5 000B6         TSTL    12(AP)                                      ;
                                                       28 13 000B9         BEQL    17$                                         ;
```

```
                     50        0C    AC D0 000BB         MOVL    SHARE_LITERAL, R0       ; 0328
                41   8F              60 91 000BF         CMPB    (R0), #65               ;
                                     05 1F 000C3         BLSSU   12$                     ;
                     50              60 9A 000C5         MOVZBL  (R0), R0                ; 0330
                                     04 11 000C8         BRB     13$                     ;
                     50        04    B0 9A 000CA 12$:    MOVZBL  @4(R0), R0              ; 0332
                53   8F              50 91 000CE 13$:    CMPB    R0, #83                 ; 0333
                                     0A 13 000D2         BEQL    16$                     ;
                                     30 DD 000D4 14$:    PUSHL   #48                     ;
         00000000G   00              01 FB 000D6 15$:    CALLS   #1, FOR$$SIGNAL_STO     ;
                                        04 000DD         RET                             ;
                17   A6        4B    8F 90 000DE 16$:    MOVB    #75, 23(FAB)            ; 0335
                     04              6C 91 000E3 17$:    CMPB    (AP), #4                ; 0342
                                     0A 1F 000E6         BLSSU   18$                     ;
                               10    AC D5 000E8         TSTL    16(AP)                  ;
                                     05 13 000EB         BEQL    18$                     ;
                36   AB        10    BC 90 000ED         MOVB    @NUM_BUF, 54(CCB)       ;
                     05              6C 91 000F2 18$:    CMPB    (AP), #5                ; 0349
                                     11 1F 000F5         BLSSU   19$                     ;
                               14    AC D5 000F7         TSTL    20(AP)                  ;
                                     0C 13 000FA         BEQL    19$                     ;
                     50        14    BC 3C 000FC         MOVZWL  @INIT_ALLOC, R0         ; 0352
                10   A6              50 D0 00100         MOVL    R0, 16(FAB)             ;
                06   A6              20 88 00104         BISB2   #32, 6(FAB)             ; 0353
                     06              6C 91 00108 19$:    CMPB    (AP), #6                ; 0360
                                     0D 1F 0010B         BLSSU   20$                     ;
                               18    AC D5 0010D         TSTL    24(AP)                  ;
                                     08 13 00110         BEQL    20$                     ;
                     50        18    BC 3C 00112         MOVZWL  @EXTEND_BLK_CNT, R0     ;
                14   A6              50 B0 00116         MOVW    R0, 20(FAB)             ;
                        00000000G    00 16 0011A 20$:    JSB     FOR$$CB_POP             ; 0366
                                        04 00120         RET                             ; 0368
                                      0000 00121 21$:    .WORD   Save nothing            ; 0209
                     50        08    AC D0 00123         MOVL    8(AP), R0               ;
                     50        04    A0 D0 00127         MOVL    4(R0), R0               ;
                               FC    A0 9F 0012B         PUSHAB  L_UNWIND_ACTION         ;
                                     01 DD 0012E         PUSHL   #1                      ;
                                     5E DD 00130         PUSHL   SP                      ;
                     7E        04    AC 7D 00132         MOVQ    4(AP), -(SP)            ;
         00000000G   00              03 FB 00136         CALLS   #3, FOR$$ERR_OPECLO     ;
                                        04 0013D         RET                             ;

; Routine Size:  318 bytes,    Routine Base:  _FOR$CODE + 0008


;  306          0369  1
;  307          0370  1 END                          ! End of module
;  308          0371  0 ELUDOM



                                       PSECT SUMMARY
;
;       Name                        Bytes                            Attributes
```

```
;
;  _FOR$CODE                         326  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        20         0        581        00:01.0
;  _$255$DUA28:[FORRTL.OBJ]FORLIB.L32;1                 711       186        26         52        00:00.5
;  _$255$DUA28:[FORRTL.OBJ]RTLLIB.L32;1                  36         0         0          8        00:00.1




;                                            COMMAND QUALIFIERS

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

; Size:         318 code + 8 data bytes
; Run Time:        00:09.3
; Elapsed Time:    00:25.5
; Lines/CPU Min:    2388
; Lexemes/CPU-Min: 13918
; Memory Used:  156 pages
; Compilation Complete
```

0178 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY