```
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
FFFFFFFFFFF       000         000 RRRRRRRRRRRR    RRRRRRRRRRRR          TTT        LLL
FFFFFFFFFFF       000         000 RRRRRRRRRRRR    RRRRRRRRRRRR          TTT        LLL
FFFFFFFFFFF       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
```

FILEID**FORENDFIL

```
FFFFFFFFFF    000000    RRRRRRRR    EEEEEEEEEE  NN      NN  DDDDDDDD    FFFFFFFFFF    IIIIII    LL
FFFFFFFFFF    000000    RRRRRRRR    EEEEEEEEEE  NN      NN  DDDDDDDD    FFFFFFFFFF    IIIIII    LL
FF          00      00  RR      RR  EE          NN      NN  DD      DD  FF              II      LL
FF          00      00  RR      RR  EE          NN      NN  DD      DD  FF              II      LL
FF          00      00  RR      RR  EE          NNNN    NN  DD      DD  FF              II      LL
FF          00      00  RR      RR  EE          NNNN    NN  DD      DD  FF              II      LL
FFFFFFFF    00      00  RRRRRRRR    EEEEEEEE    NN  NN  NN  DD      DD  FFFFFFFF        II      LL
FFFFFFFF    00      00  RRRRRRRR    EEEEEEEE    NN  NN  NN  DD      DD  FFFFFFFF        II      LL
FF          00      00  RR  RR      EE          NN    NNNN  DD      DD  FF              II      LL
FF          00      00  RR  RR      EE          NN    NNNN  DD      DD  FF              II      LL
FF          00      00  RR    RR    EE          NN      NN  DD      DD  FF              II      LL
FF          00      00  RR    RR    EE          NN      NN  DD      DD  FF              II      LL
FF            000000    RR      RR  EEEEEEEEEE  NN      NN  DDDDDDDD    FF            IIIIII    LLLLLLLLLL
FF            000000    RR      RR  EEEEEEEEEE  NN      NN  DDDDDDDD    FF            IIIIII    LLLLLLLLLL

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 FOR$ENDFILE (    ! FORTRAN ENDFILE statement
;     2          0002  0         IDENT = '1-006' ! File: FORENDFIL.B32  Edit: SBL1006
;     3          0003  0                 ) =
;     4          0004  1 BEGIN
;     5          0005  1
;     6          0006  1 !
;     7          0007  1 !*****************************************************************************
;     8          0008  1 !*                                                                           *
;     9          0009  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
;    10          0010  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                    *
;    11          0011  1 !*  ALL RIGHTS RESERVED.                                                     *
;    12          0012  1 !*                                                                           *
;    13          0013  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED     *
;    14          0014  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE    *
;    15          0015  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER     *
;    16          0016  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY     *
;    17          0017  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY     *
;    18          0018  1 !*  TRANSFERRED.                                                              *
;    19          0019  1 !*                                                                           *
;    20          0020  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE     *
;    21          0021  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT     *
;    22          0022  1 !*  CORPORATION.                                                              *
;    23          0023  1 !*                                                                           *
;    24          0024  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS     *
;    25          0025  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                   *
;    26          0026  1 !*                                                                           *
;    27          0027  1 !*                                                                           *
;    28          0028  1 !*****************************************************************************
;    29          0029  1 !
;    30          0030  1
;    31          0031  1 !++
;    32          0032  1 ! FACILITY:  FORTRAN Support Library
;    33          0033  1 !
;    34          0034  1 ! ABSTRACT:
;    35          0035  1 !
;    36          0036  1 !       Contains routine FOR$ENDFILE for support of FORTRAN endfile
;    37          0037  1 !       statement.
;    38          0038  1 !
;    39          0039  1 ! ENVIRONMENT:  User Mode - AST re-entrant
;    40          0040  1 !
;    41          0041  1 ! AUTHOR:  Jonathan M. Taylor, CREATION DATE: 24-Oct-1977
;    42          0042  1 !
;    43          0043  1 ! MODIFIED BY:
;    44          0044  1 !
;    45          0045  1 ! [Previous edit history deleted.  SBL 15-July-1981]
;    46          0046  1 ! 1-001 - Update version number and copyright notice.  JBS 16-NOV-78
;    47          0047  1 ! 1-002 - Change REQUIRE file names from FOR... to OTS...  JBS 07-DEC-78
;    48          0048  1 ! 1-003 - Change OTSOPN back to FOROPN, and change the prefix on LUN
;    49          0049  1 !         parameters from OPEN to LUB.  JBS 13-DEC-78
;    50          0050  1 ! 1-004 - Implement ERR= and IOSTAT=.  SBL 1-May-1979
;    51          0051  1 ! 1-005 - Correct module name.  SBL 15-July-1981
;    52          0052  1 ! 1-006 - Move ACTUALCOUNT declaration inside routine.  Add SWITCHES
;    53          0053  1 !         declaration.  SBL 16-Jun-1982
;    54          0054  1 !--
```

```
;    56          0055  1 !
;    57          0056  1 ! SWITCHES:
;    58          0057  1 !
;    59          0058  1
;    60          0059  1 SWITCHES ADDRESSING_MODE (EXTERNAL = GENERAL, NONEXTERNAL = WORD_RELATIVE);
;    61          0060  1
;    62          0061  1 !
;    63          0062  1 ! LINKAGES:
;    64          0063  1 !
;    65          0064  1 REQUIRE 'RTLIN:OTSLNK';         ! define all linkages
;    66          0493  1
;    67          04    1 !
;    68          0495  1 ! TABLE OF CONTENTS:
;    69          0496  1 !
;    70          0497  1
;    71          0498  1 FORWARD ROUTINE
;    72          0499  1         FOR$ENDFILE;
;    73          0500  1
;    74          0501  1 !
;    75          0502  1 ! INCLUDE FILES:
;    76          0503  1 !
;    77          0504  1 REQUIRE 'RTLML:FORERR';         ! FORTRAN error number definitions
;    78          0572  1 REQUIRE 'RTLIN:OTSMAC';         ! Define macros
;    79          0766  1 REQUIRE 'RTLIN:RTLPSECT',       ! Define DECLARE_PSECTS macro
;    80          0861  1 LIBRARY 'RTLSTARLE';            ! STARLET library for symbols and macros
;    81          0862  1 REQUIRE 'RTLIN:FOROPN';         ! FORTRAN OPEN parameters
;    82          1125  1 REQUIRE 'RTLML:FORPAR';         ! inter-module constants
;    83          1148  1 REQUIRE 'RTLML:OTSLUB';         ! LUB offset definitions
;    84          1288  1 REQUIRE 'RTLML:OTSISB';         ! to get ISB length only
;    85          1456  1 !
;    86          1457  1 ! MACROS:
;    87          1458  1 !       NONE
;    88          1459  1
;    89          1460  1 !
;    90          1461  1 ! EQUATED SYMBOLS:
;    91          1462  1 !       NONE
;    92          1463  1
;    93          1464  1 !
;    94          1465  1 ! PSECT DECLARATIONS:
;    95          1466  1 !
;    96          1467  1
;    97          1468  1         DECLARE_PSECTS (FOR);          ! declare PSECTs for FOR$ facility
;    98          1469  1
;    99          1470  1 !
;   100          1471  1 ! OWN STORAGE:
;   101          1472  1 !       NONE
;   102          1473  1
;   103          1474  1 !
;   104          1475  1 ! EXTERNAL REFERENCES:
;   105          1476  1 !
;   106          1477  1
;   107          1478  1 EXTERNAL ROUTINE
;   108          1479  1         FOR$$IOSTAT_HND,                        ! error handler
;   109          1480  1         FOR$$CB_PUSH: JSB_CB_PUSH NOVALUE,              ! Get a LUB/ISB/RAB
;   110          1481  1         FOR$$CB_POP: JSB_CB_POP NOVALUE,                ! Return the I/O system
;   111          1482  1         FOR$$SIGNAL_STO: NOVALUE,       ! convert error number and signal_stop
;   112          1483  1         FOR$$OPEN_DEFLT: CALL_CCB NOVALUE;      ! default OPEN routine
```

```
;  114          1484  1 GLOBAL ROUTINE FOR$ENDFILE  (    ! FORTRAN ENDFILE statement support
;  115          1485  1         UNIT,
;  116          1486  1         ERR_EQL)
;  117          1487  1         =
;  118          1488  1 !++
;  119          1489  1 ! FUNCTIONAL DESCRIPTION:
;  120          1490  1 !
;  121          1491  1 !       Perform FORTRAN ENDFILE semantics.  Check file and record type
;  122          1492  1 !       for legality of the ENDFILE.
;  123          1493  1 !       Open the file if not already open and set flag to finish defaults later.
;  124          1494  1 !       Write a one byte ^Z record.
;  125          1495  1 !
;  126          1496  1 ! FORMAL PARAMETERS:
;  127          1497  1 !
;  128          1498  1 !       UNIT.rlu.v                      logical unit number
;  129          1499  1 !       ERR_EQL.rl.v                    If 0 or omitted, all errors are signalled.
;  130          1500  1 !                                       If non-zero, errors unwind to caller.
;  131          1501  1 !
;  132          1502  1 ! IMPLICIT INPUTS:
;  133          1503  1 !
;  134          1504  1 !       LUB$V_NOTSEQORG                 1 if ORGANIZATION is not 'SEQUENTIAL'
;  135          1505  1 !       LUB$V_DIRECT                    1 if ACCESS = 'DIRECT' or define file done
;  136          1506  1 !       LUB$V_FIXED                     1 if fixed length records specified in open
;  137          1507  1 !       LUB$V_UNFORMAT                  1 if FORMAT = 'UNFORMATTED' or define
;  138          1508  1 !                                       file done
;  139          1509  1 !       LUB$V_SEGMENTED                 1 if SEGMENTED RECORDTYPE in OPEN statement
;  140          1510  1 !                                       or defaulted that way
;  141          1511  1 !       LUB$V_OPENED                    1 if file already open on this unit
;  142          1512  1 !
;  143          1513  1 ! IMPLICIT OUTPUTS:
;  144          1514  1 !
;  145          1515  1 !       LUB$L_LOG_RECNO                 Incremented if ^Z record written
;  146          1516  1 !
;  147          1517  1 ! ROUTINE VALUE:
;  148          1518  1 !
;  149          1519  1 !       An IOSTAT value.
;  150          1520  1 !
;  151          1521  1 ! SIDE EFFECTS:
;  152          1522  1 !
;  153          1523  1 !       File is opened if closed previous to ENDFILE
;  154          1524  1 !       SIGNAL_STOPs error FOR$_ENDFILERR if ENDFILE ignored for this type file
;  155          1525  1 !       SIGNAL_STOPs FOR$_ENDFILERR if RMS $PUT error.
;  156          1526  1 !
;  157          1527  1 !--
;  158          1528  1
;  159          1529  2     BEGIN
;  160          1530  2
;  161          1531  2     GLOBAL REGISTER
;  162          1532  2         CCB = 11: REF BLOCK[, BYTE];
;  163          1533  2
;  164          1534  2     LOCAL
;  165          1535  2         L_UNWIND_ACTION: VOLATILE,
;  166          1536  2         L_ERR_EQL_PRES: VOLATILE,
;  167          1537  2         L_EOF_CHAR: BYTE;
;  168          1538  2
;  169          1539  2     BUILTIN
;  170          1540  2         ACTUALCOUNT;
```

```
;  171          1541  2
;  172          1542  2      ENABLE
;  173          1543  2          FOR$$IOSTAT_HND (L_UNWIND_ACTION, L_ERR_EQL_PRES);
;  174          1544  2
;  175          1545  2      !+
;  176          1546  2      ! Determine if ERR= is present.
;  177          1547  2      !-
;  178          1548  2
;  179          1549  2      IF ACTUALCOUNT () GTR 1
;  180          1550  2      THEN
;  181          1551  2          L_ERR_EQL_PRES = .ERR_EQL
;  182          1552  2      ELSE
;  183          1553  2          L_ERR_EQL_PRES = 0;
;  184          1554  2
;  185          1555  2      !+
;  186          1556  2      ! Action on error is not to pop a LUB.
;  187          1557  2      !-
;  188          1558  2
;  189          1559  2      L_UNWIND_ACTION = FOR$K_UNWINDNOP;
;  190          1560  2
;  191          1561  2      !+
;  192          1562  2      ! Allocate a LUB/ISB/RAB for this logical unit.
;  193          1563  2      ! ON return, CCB points to the current control block.
;  194          1564  2      !-
;  195          1565  2
;  196          1566  2      FOR$$CB_PUSH (.UNIT, LUB$K_LUN_MIN);
;  197          1567  2
;  198          1568  2      !+
;  199          1569  2      ! UNwind action is now to pop the LUB.
;  200          1570  2      !-
;  201          1571  2
;  202          1572  2      L_UNWIND_ACTION = FOR$K_UNWINDPOP;
;  203          1573  2
;  204          1574  2      !+
;  205          1575  2      ! Check various file attributes.  File must be:
;  206          1576  2      !   Sequential organization
;  207          1577  2      !   Sequential access
;  208          1578  2      !   Variable length records
;  209          1579  2      !   Must have segmented records if unformatteed
;  210          1580  2      !-
;  211          1581  2
;  212          1582  2      IF .CCB[LUB$V_NOTSEQORG] OR
;  213          1583  2          .CCB[LUB$V_DIRECT] OR
;  214          1584  2          .CCB[LUB$V_FIXED] OR
;  215          1585  3          (.CCB[LUB$V_UNFORMAT] AND .CCB[LUB$V_SEGMENTED] EQL 0)
;  216          1586  2      THEN
;  217          1587  2          FOR$$SIGNAL_STO (FOR$K_ENDFILERR)
;  218          1588  2      ELSE
;  219          1589  2
;  220          1590  3          BEGIN
;  221          1591  3
;  222          1592  3          !+
;  223          1593  3          ! If the file is not open, default open it.
;  224          1594  3          ! But do not specify whether formatted or unformatted.
;  225          1595  3          ! That will depend on the next I/O statement executed
;  226          1596  3          ! so just set LUB$V_ENDFILOPN to be tested on every I/O statement.
;  227          1597  3          ! When set, it will be cleared and the remaining defaults will be specified.
```

```
;  228          1598  3          !-
;  229          1599  3
;  230          1600  3          IF .CCB[LUB$V_OPENED] EQL 0
;  231          1601  3          THEN
;  232          1602  4              BEGIN
;  233          1603  4              FOR$$OPEN_DEFLT (OPEN$K_ACC_SEQ,
;  234          1604  4                               OPEN$K_TYP_NEW,
;  235          1605  4                               OPEN$K_FOR_UNS);
;  236          1606  4              CCB[LUB$V_ENDFILOPN] = T;
;  237          1607  3              END;
;  238          1608  3
;  239          1609  3          !+
;  240          1610  3          ! Manually write a one byte control Z record to the file.
;  241          1611  3          !-
;  242          1612  3
;  243          1613  3          L_EOF_CHAR = FOR$K_CONTROL_Z;
;  244          1614  3          CCB[RAB$L_RBF] = L_EOF_CHAR;
;  245          1615  3          CCB[RAB$W_RSZ] = 1;
;  246          1616  3
;  247          1617  4          IF NOT $PUT (RAB = .CCB)
;  248          1618  3          THEN
;  249          1619  3              FOR$$SIGNAL_STO (FOR$K_ENDFILERR);
;  250          1620  3
;  251          1621  3          !+
;  252          1622  3          ! Increment the logical record number for backspace.
;  253          1623  3          !-
;  254          1624  3
;  255          1625  3          CCB[LUB$L_LOG_RECNO] = .CCB[LUB$L_LOG_RECNO] + 1;
;  256          1626  3
;  257          1627  2          END;
;  258          1628  2
;  259          1629  2      !+
;  260          1630  2      ! Return the I/O system to its former state.
;  261          1631  2      !-
;  262          1632  2
;  263          1633  2      FOR$$CB_POP ();
;  264          1634  2
;  265          1635  2      RETURN 0;           ! Success IOSTAT value
;  266          1636  1      END;


                                                        .TITLE  FOR$ENDFILE
                                                        .IDENT  \1-006\

                                                        .EXTRN  FOR$$IOSTAT_HND
                                                        .EXTRN  FOR$$CB_PUSH, FOR$$CB_POP
                                                        .EXTRN  FOR$$SIGNAL_STO
                                                        .EXTRN  FOR$$OPEN_DEFLT
                                                        .EXTRN  SYS$PUT

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

                                    080C 00000          .ENTRY  FOR$ENDFILE, Save R2,R3,R11                     ; 1484
                53 00000000G    00   9E 00002          MOVAB   FOR$$SIGNAL_STO, R3                             ;
                5E                0C   C2 00009          SUBL2   #12, SP                                         ;
                        04        AE   7C 0000C          CLRQ    L_ERR_EQL_PRES                                  ; 1529
                6D      0081      CF   DE 0000F          MOVAL   8$, (FP)                                        ;
```

```
                        01              6C  91 00014           CMPB    (AP), #1                        ; 1549
                                        07  1B 00017           BLEQU   1$
                    04  AE      08      AC  D0 00019           MOVL    ERR_EQL, L_ERR_EQL_PRES         ; 1551
                                        03  11 0001E           BRB     2$
                            04  AE      D4 00020 1$:           CLRL    L_ERR_EQL_PRES                  ; 1553
                    08  AE              01  D0 00023 2$:       MOVL    #1, L_UNWIND_ACTION             ; 1559
                                        50  D4 00027           CLRL    R0                              ; 1566
                        52          04  AC  D0 00029           MOVL    UNIT, R2
                            000000000G  00  16 0002D           JSB     FOR$$CB_PUSH
                                    08  AE  D4 00033           CLRL    L_UNWIND_ACTION                 ; 1572
                14  A1  AB              03  E0 00036           BBS     #3, -95(CCB), 3$                ; 1582
                0F  FC  AB              04  E0 0003B           BBS     #4, -4(CCB), 3$                 ; 1583
                0A  FD  AB              02  E0 00040           BBS     #2, -3(CCB), 3$                 ; 1584
                0C  FD  AB              01  E1 00045           BBC     #1, -3(CCB), 4$                 ; 1585
                07  FD  AB              03  E0 0004A           BBS     #3, -3(CCB), 4$
                                        21  DD 0004F 3$:       PUSHL   #33                             ; 1587
                        63              01  FB 00051           CALLS   #1, FOR$$SIGNAL_STO
                                        35  11 00054           BRB     7$
                        12          FC  AB  E8 00056 4$:       BLBS    -4(CCB), 5$                     ; 1600
                        7E              01  CE 0005A           MNEGL   #1, -(SP)                       ; 1603
                                        02  DD 0005D           PUSHL   #2
                                        02  DD 0005F           PUSHL   #2
            000000000G  00              03  FB 00061           CALLS   #3, FOR$$OPEN_DEFLT
                    FE  AB              02  88 00068           BISB2   #2, -2(CCB)                     ; 1606
                    6E                  1A  90 0006C 5$:       MOVB    #26, L_EOF_CHAR                 ; 1613
                    28  AB              6E  9E 0006F           MOVAB   L_EOF_CHAR, 40(CCB)             ; 1614
                    22  AB              01  B0 00073           MOVW    #1, 34(CCB)                     ; 1615
                                        5B  DD 00077           PUSHL   CCB                             ; 1617
            000000000G  00              01  FB 00079           CALLS   #1, SYS$PUT
                        05              50  E8 00080           BLBS    R0, 6$
                                        21  DD 00083           PUSHL   #33                             ; 1619
                        63              01  FB 00085           CALLS   #1, FOR$$SIGNAL_STO
                                E0  AB      D6 00088 6$:       INCL    -32(CCB)                        ; 1625
                            000000000G  00  16 0008B 7$:       JSB     FOR$$CB_POP                     ; 1633
                                        50  D4 00091           CLRL    R0                              ; 1635
                                            04 00093           RET                                     ; 1636
                                          0000 00094 8$:       .WORD   Save nothing                    ; 1529
                        50          08  AC  D0 00096           MOVL    8(AP), R0
                        50          04  A0  D0 0009A           MOVL    4(R0), R0
                                    F8  A0  9F 0009E           PUSHAB  L_ERR_EQL_PRES
                                    FC  A0  9F 000A1           PUSHAB  L_UNWIND_ACTION
                                        02  DD 000A4           PUSHL   #2
                                        5E  DD 000A6           PUSHL   SP
                        7E          04  AC  7D 000A8           MOVQ    4(AP), -(SP)
            000000000G  00              03  FB 000AC           CALLS   #3, FOR$$IOSTAT_HND
                                            04 000B3           RET

; Routine Size:  180 bytes,    Routine Base:  _FOR$CODE + 0000

;  267          1637  1
;  268          1638  1 END
;  269          1639  0 ELUDOM
```

```
;                                   PSECT SUMMARY
;
;       Name                    Bytes                           Attributes
;
;  _FOR$CODE                        180  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         7         0         581        00:01.1
;
;
;
;
;                                   COMMAND QUALIFIERS
;
;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACE/LIS=LIS$:FORENDFIL/OBJ=OBJ$:FORENDFIL MSRC$:FORENDFIL/UPDATE=(ENH$:FORENDFIL
;       )
;
; Size:          180 code + 0 data bytes
; Run Time:        00:12.9
; Elapsed Time:    00:47.2
; Lines/CPU Min:    7652
; Lexemes/CPU-Min: 39646
; Memory Used:  168 pages
; Compilation Complete
```

FORFIND
LIS

FORFMTINT
LIS

FOREXITHA
LIS

FORENODEF
LIS

FORENCOMF
LIS

FORIDATE
LIS

FORENDFIL
LIS

FORERRSNS
LIS

FOREXIT
LIS

FORERROR
LIS

FORFMTCP
LIS

FORENCOMO
LIS

FORTNIDES
LIS