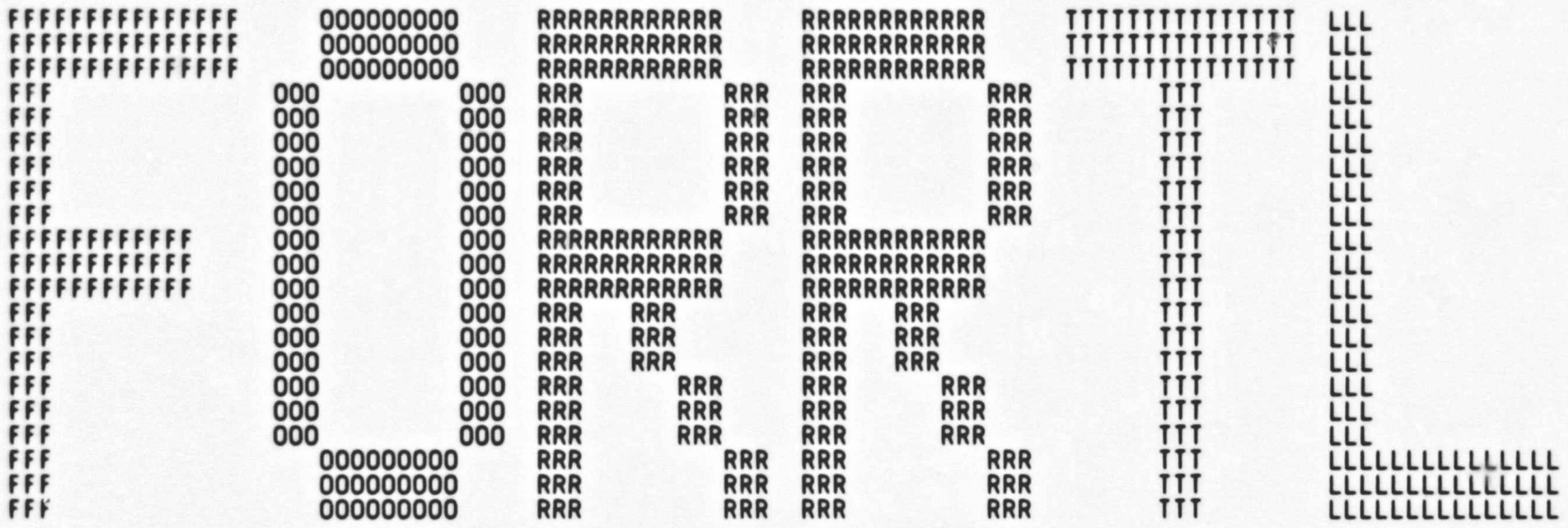
FORTRAN
LLLLLLLLLLLLLLL

FILEID**FORREWIND

```
FFFFFFFFFF    000000    RRRRRRRR    RRRRRRRR    EEEEEEEEEE  WW      WW    IIIIII    NN      NN  DDDDDDDD
FFFFFFFFFF    000000    RRRRRRRR    RRRRRRRR    EEEEEEEEEE  WW      WW    IIIIII    NN      NN  DDDDDDDD
FF          00      00  RR      RR  RR      RR  EE          WW      WW      II      NN      NN  DD      DD
FF          00      00  RR      RR  RR      RR  EE          WW      WW      II      NN      NN  DD      DD
FF          00      00  RR      RR  RR      RR  EE          WW      WW      II      NNNN    NN  DD      DD
FF          00      00  RR      RR  RR      RR  EE          WW      WW      II      NNNN    NN  DD      DD
FFFFFFFF    00      00  RRRRRRRR    RRRRRRRR    EEEEEEEE    WW      WW      II      NN  NN  NN  DD      DD
FFFFFFFF    00      00  RRRRRRRR    RRRRRRRR    EEEEEEEE    WW      WW      II      NN  NN  NN  DD      DD
FF          00      00  RR  RR      RR  RR      EE          WW  WW  WW      II      NN    NNNN  DD      DD
FF          00      00  RR  RR      RR  RR      EE          WW  WW  WW      II      NN    NNNN  DD      DD
FF          00      00  RR    RR    RR    RR    EE          WWWW  WWWW      II      NN      NN  DD      DD    ....
FF          00      00  RR    RR    RR    RR    EE          WWWW  WWWW      II      NN      NN  DD      DD    ....
FF            000000    RR      RR  RR      RR  EEEEEEEEEE  WW      WW    IIIIII    NN      NN  DDDDDDDD    ....
FF            000000    RR      RR  RR      RR  EEEEEEEEEE  WW      WW    IIIIII    NN      NN  DDDDDDDD    ....

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$REWIND (      ! FORTRAN REWIND Statement
;     2          0002  0         IDENT = '1-007' ! File: FORREWIND.B32   ! Edit SBL1007
;     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, user callable
;    33          0033  1 !
;    34          0034  1 ! ABSTRACT:
;    35          0035  1 !
;    36          0036  1 !       Contains routine FOR$REWIND: rewind a FORTRAN sequential
;    37          0037  1 !       access file.
;    38          0038  1 !
;    39          0039  1 ! ENVIRONMENT: Mixture of AST level or not.
;    40          0040  1 !
;    41          0041  1 ! AUTHOR: Jonathan M. Taylor, CREATION DATE: 10-OCT-77
;    42          0042  1 !
;    43          0043  1 ! MODIFIED BY:
;    44          0044  1 !
;    45          0045  1 !       Jonathan M. Taylor, 10-OCT-77 : VERSION 0
;    46          0046  1 ! Previous edit history removed.  SBL 16-June-1982
;    47          0047  1 ! 1-001 - Update version number and copyright notice.  JBS 16-NOV-78
;    48          0048  1 ! 1-002 - Change REQUIRE file names from FOR... to OTS...  JBS 06-DEC-78
;    49          0049  1 ! 1-003 - Change prefix of LUN literals 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 - Error instead of no-op on not open or direct.  SBL 2-May-1979
;    52          0052  1 ! 1-006 - 1-005 is a mistake.  No-op if not open, error if not
;    53          0053  1 !         sequential org and access.  SBL 16-May-1979
;    54          0054  1 ! 1-007 - Allow errors RMS$_IOP, RMS$_BOF and RMS$_EOF from $REWIND.
;    55          0055  1 !         Move declaration of ACTUALCOUNT.  Add SWITCHES.  SBL 16-June-1982
;    56          0056  1 !--
```

```
;    58          0057  1 !
;    59          0058  1 ! SWITCHES:
;    60          0059  1 !
;    61          0060  1
;    62          0061  1 SWITCHES ADDRESSING_MODE (EXTERNAL = GENERAL, NONEXTERNAL = WORD_RELATIVE);
;    63          0062  1
;    64          0063  1 !
;    65          0064  1 ! LINKAGES:
;    66          0065  1 !
;    67          0066  1 REQUIRE 'RTLIN:OTSLNK';         ! Define all linkages
;    68          0495  1
;    69          0496  1 !
;    70          0497  1 ! TABLE OF CONTENTS:
;    71          0498  1 !
;    72          0499  1
;    73          0500  1 FORWARD ROUTINE
;    74          0501  1         FOR$REWIND;                     ! FORTRAN REWIND statement processor
;    75          0502  1
;    76          0503  1 !
;    77          0504  1 ! INCLUDE FILES:
;    78          0505  1 !
;    79          0506  1
;    80          0507  1 REQUIRE 'RTLML:FORERR';                 ! FORTRAN error number definitions
;    81          0575  1 REQUIRE 'RTLML:OTSLUB';                 ! Logical Unit Block definitions
;    82          0715  1 REQUIRE 'RTLIN:OTSMAC';                 ! Macros
;    83          0909  1 REQUIRE 'RTLIN:RTLPSECT';               ! Define DECLARE_PSECTS macro
;    84          1004  1 REQUIRE 'RTLML:OTSISB';                 ! ISB definitions
;    85          1172  1 REQUIRE 'RTLML:FORPAR';                 ! FORTRAN inter-module parameters
;    86          1195  1         LIBRARY 'RTLSTARLE';            ! STARLET library for macros and symbols
;    87          1196  1 !
;    88          1197  1 ! MACROS:
;    89          1198  1 !       NONE
;    90          1199  1
;    91          1200  1 !
;    92          1201  1 ! EQUATED SYMBOLS:
;    93          1202  1 !       NONE
;    94          1203  1
;    95          1204  1
;    96          1205  1 !
;    97          1206  1 ! OWN STORAGE:
;    98          1207  1 !       NONE
;    99          1208  1
;   100          1209  1 !
;   101          1210  1 ! EXTERNAL REFERENCES:
;   102          1211  1 !
;   103          1212  1 EXTERNAL ROUTINE
;   104          1213  1         FOR$$IOSTAT_HND,                ! error condition handler
;   105          1214  1         FOR$$SIGNAL_STO: NOVALUE,       ! convert error number and signal
;   106          1215  1         FOR$$CB_PUSH: JSB_CB_PUSH NOVALUE,      ! create LUB/ISB/RAB, if needed
;   107          1216  1         FOR$$CB_POP: JSB_CB_POP NOVALUE;        ! return I/O system to previous state
;   108          1217  1
;   109          1218  1 !
;   110          1219  1 ! PSECT DECLARATIONS:
;   111          1220  1 !
;   112          1221  1
;   113          1222  1         DECLARE_PSECTS (FOR);           ! declare PSECTS for FOR$ facility
```

```
;  115          1223  1 GLOBAL ROUTINE FOR$REWIND (
;  116          1224  1         UNIT,
;  117          1225  1         ERR_EQL)
;  118          1226  1         =
;  119          1227  1
;  120          1228  1 !++
;  121          1229  1 ! FUNCTIONAL DESCRIPTION:
;  122          1230  1 !
;  123          1231  1 !       Perform RMS rewind operation on the file specified by the
;  124          1232  1 !       UNIT parameter.
;  125          1233  1 !
;  126          1234  1 ! FORMAL PARAMETERS:
;  127          1235  1 !
;  128          1236  1 !       UNIT.rlu.v              Logical unit number
;  129          1237  1 !       ERR_EQL.rl.v            If 0 or not present, signal errors
;  130          1238  1 !                               If non-zero, unwind to caller.
;  131          1239  1 !
;  132          1240  1 ! IMPLICIT INPUTS:
;  133          1241  1 !
;  134          1242  1 !       LUB$V_DIRECT            This unit has previously been specified
;  135          1243  1 !                               for direct access by an OPEN statement or
;  136          1244  1 !                               DEFINE FILE.
;  137          1245  1 !       LUB$V_OPENED            This unit has already been opened by
;  138          1246  1 !                               an OPEN statement or default open.
;  139          1247  1 !
;  140          1248  1 ! IMPLICIT OUTPUTS:
;  141          1249  1 !
;  142          1250  1 !       LUB$L_LOG_RECNO         set to 1.
;  143          1251  1 !
;  144          1252  1 ! ROUTINE VALUE:
;  145          1253  1 !
;  146          1254  1 !       An IOSTAT value.
;  147          1255  1 !
;  148          1256  1 ! SIDE EFFECTS:
;  149          1257  1 !
;  150          1258  1 !       SIGNAL_STOPs FOR$_REWERR if a non-EOF error is returned from
;  151          1259  1 !       the RMS rewind call.
;  152          1260  1 !
;  153          1261  1 !--
;  154          1262  1
;  155          1263  2     BEGIN
;  156          1264  2
;  157          1265  2     GLOBAL REGISTER
;  158          1266  2         CCB = 11: REF BLOCK[, BYTE];
;  159          1267  2
;  160          1268  2     LOCAL
;  161          1269  2         STATUS,                             ! Return status from $REWIND
;  162          1270  2         L_UNWIND_ACTION: VOLATILE,          ! Unwind action code (FOR$K_UNWIND{POP or NOP})
;  163          1271  2         L_ERR_EQL_PRES: VOLATILE;           ! 1 if ERR= present
;  164          1272  2
;  165          1273  2     BUILTIN
;  166          1274  2         ACTUALCOUNT;
;  167          1275  2
;  168          1276  2     ENABLE
;  169          1277  2         FOR$$IOSTAT_HND (L_UNWIND_ACTION, L_ERR_EQL_PRES);
;  170          1278  2                                             ! pass info to error handler
;  171          1279  2
```

```
;  172          1280  2      !+
;  173          1281  2      ! Determine if ERR= is present.
;  174          1282  2      !-
;  175          1283  2
;  176          1284  2      IF ACTUALCOUNT () GTR 1
;  177          1285  2      THEN
;  178          1286  2          L_ERR_EQL_PRES = .ERR_EQL
;  179          1287  2      ELSE
;  180          1288  2          L_ERR_EQL_PRES = 0;
;  181          1289  2
;  182          1290  2      !+
;  183          1291  2      ! Set up error handler conditions in case CB_PUSH bombs
;  184          1292  2      !-
;  185          1293  2
;  186          1294  2      L_UNWIND_ACTION = FOR$K_UNWINDNOP;
;  187          1295  2
;  188          1296  2      !+
;  189          1297  2      ! Get a LUB for this logical unit.
;  190          1298  2      ! On return, CCB points to the current control block.
;  191          1299  2      !-
;  192          1300  2
;  193          1301  2      FOR$$CB_PUSH (.UNIT, LUB$K_LUN_MIN);
;  194          1302  2
;  195          1303  2      !+
;  196          1304  2      ! Unwind action (if an error occurs) is now to pop a LUB.
;  197          1305  2      !-
;  198          1306  2
;  199          1307  2      L_UNWIND_ACTION = FOR$K_UNWINDPOP;
;  200          1308  2
;  201          1309  2      !+
;  202          1310  2      ! Check the LUB.  If file is not open, then this is a no-op.
;  203          1311  2      ! Else must be sequential organization and access.
;  204          1312  2      !-
;  205          1313  2
;  206          1314  2      IF .CCB [LUB$V_OPENED]
;  207          1315  2      THEN
;  208          1316  2          IF NOT .CCB [LUB$V_DIRECT] AND NOT .CCB [LUB$V_NOTSEQORG]
;  209          1317  2          THEN
;  210          1318  3              BEGIN
;  211          1319  3
;  212          1320  3              !+
;  213          1321  3              ! Call RMS to REWIND the file, all failure codes returned
;  214          1322  3              ! cause a SIGNAL_STOP to occur, except for IOP, EOF or BOF.
;  215          1323  3              !-
;  216          1324  3
;  217          1325  4              IF NOT (STATUS = $REWIND (RAB = .CCB))
;  218          1326  3              THEN
;  219          1327  4                  BEGIN
;  220          1328  4                  IF .STATUS NEQ RMS$_IOP AND
;  221          1329  4                     .STATUS NEQ RMS$_EOF AND
;  222          1330  4                     .STATUS NEQ RMS$_BOF
;  223          1331  4                  THEN
;  224          1332  5                      BEGIN
;  225          1333  5                      FOR$$SIGNAL_STO (FOR$K_REWERR);
;  226          1334  5                      RETURN 0;
;  227          1335  4                      END;
;  228          1336  3                  END;
```

```
;  229          1337  3              !+
;  230          1338  3              ! Clear APPEND flag - OK for backspace now
;  231          1339  3              !-
;  232          1340  3
;  233          1341  3              CCB[LUB$V_APPEND] = 0;
;  234          1342  3
;  235          1343  3              !+
;  236          1344  3              ! Set the logical record number to 1.
;  237          1345  3              !-
;  238          1346  3
;  239          1347  3              CCB[LUB$L_LOG_RECNO] = 1;
;  240          1348  3
;  241          1349  3              END
;  242          1350  3          ELSE
;  243          1351  2              BEGIN
;  244          1352  3              FOR$$SIGNAL_STO (FOR$K_REWERR);
;  245          1353  3              RETURN 0;
;  246          1354  3              END;
;  247          1355  2
;  248          1356  2      !+
;  249          1357  2      ! Return the file system to its former state.
;  250          1358  2      !-
;  251          1359  2
;  252          1360  2      FOR$$CB_POP ();
;  253          1361  2      RETURN 0;           ! Success IOSTAT value
;  254          1362  2      END;
;  255          1363  1

                                                .TITLE  FOR$REWIND
                                                .IDENT  \1-007\

                                                .EXTRN  FOR$$IOSTAT_HND
                                                .EXTRN  FOR$$SIGNAL_STO
                                                .EXTRN  FOR$$CB_PUSH, FOR$$CB_POP
                                                .EXTRN  SYS$REWIND

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

                                  0804 00000    .ENTRY  FOR$REWIND, Save R2,R11                 ; 1223
                          5E            04 C2 00002    SUBL2   #4, SP                          ;
                                        7E D4 00005    CLRL    L_ERR_EQL_PRES                  ; 1263
                              04        AE D4 00007    CLRL    L_UNWIND_ACTION                 ;
                          6D      0074  CF DE 0000A    MOVAL   7$, (FP)                        ;
                          01            6C 91 0000F    CMPB    (AP), #1                        ; 1284
                                        06 1B 00012    BLEQU   1$                              ;
                          6E        08  AC D0 00014    MOVL    ERR_EQL, L_ERR_EQL_PRES         ; 1286
                                        02 11 00018    BRB     2$                              ;
                                        6E D4 0001A 1$: CLRL    L_ERR_EQL_PRES                  ; 1288
                    04    AE            01 D0 0001C 2$: MOVL    #1, L_UNWIND_ACTION             ; 1294
                                        50 D4 00020    CLRL    R0                              ; 1301
                          52        04  AC D0 00022    MOVL    UNIT, R2                        ;
                              00000000G 00 16 00026    JSB     FOR$$CB_PUSH                    ;
                                    04  AE D4 0002C    CLRL    L_UNWIND_ACTION                 ; 1307
                          46        FC  AB E9 0002F    BLBC    -4(CCB), 5$                     ; 1314
                36        FC  AB        04 E0 00033    BBS     #4, -4(CCB), 4$                 ; 1316
                31        A1  AB        03 E0 00038    BBS     #3, -95(CCB), 4$                ;
```

```
                                 5B  DD 0003D        PUSHL   CCB                              ; 1325
        00000000G  00            01  FB 0003F        CALLS   #1, SYS$REWIND                   ;
                   1B            50  E8 00046        BLBS    STATUS, 3$                       ;
        00018574   8F            50  D1 00049        CMPL    STATUS, #99700                   ; 1328
                                 12  13 00050        BEQL    3$                               ;
        0001827A   8F            50  D1 00052        CMPL    STATUS, #98938                   ; 1329
                                 09  13 00059        BEQL    3$                               ;
        00018198   8F            50  D1 0005B        CMPL    STATUS, #98712                   ; 1330
                                 0A  12 00062        BNEQ    4$                               ;
              FD   AB            20  8A 00064 3$:    BICB2   #32, -3(CCB)                     ; 1342
              E0   AB            01  D0 00068        MOVL    #1, -32(CCB)                     ; 1348
                                 0B  11 0006C        BRB     5$                               ; 1316
                                 14  DD 0006E 4$:    PUSHL   #20                              ; 1353
        00000000G  00            01  FB 00070        CALLS   #1, FOR$$SIGNAL_STO              ;
                                 06  11 00077        BRB     6$                               ; 1354
                    00000000G    00  16 00079 5$:    JSB     FOR$$CB_POP                      ; 1361
                             50  D4 0007F 6$:        CLRL    R0                               ; 1363
                                 04 00081            RET                                      ;
                               0000 00082 7$:        .WORD   Save nothing                     ; 1263
                   50       08   AC  D0 00084        MOVL    8(AP), R0                        ;
                   50       04   A0  D0 00088        MOVL    4(R0), R0                        ;
                            F8   A0  9F 0008C        PUSHAB  L_ERR_EQL_PRES                   ;
                            FC   A0  9F 0008F        PUSHAB  L_UNWIND_ACTION                  ;
                                 02  DD 00092        PUSHL   #2                               ;
                                 5E  DD 00094        PUSHL   SP                               ;
                   7E       04   AC  7D 00096        MOVQ    4(AP), -(SP)                     ;
        00000000G  00            03  FB 0009A        CALLS   #3, FOR$$IOSTAT_HND              ;
                                 04 000A1            RET                                      ;

; Routine Size:  162 bytes,    Routine Base:  _FOR$CODE + 0000


;   256         1364  1
;   257         1365  1 END                                   !End of module
;   258         1366  0 ELUDOM
```

```
;                              PSECT SUMMARY
;
;       Name                         Bytes                       Attributes
;
;   _FOR$CODE                            162  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       00:01.3
```

```
;                                       COMMAND QUALIFIERS

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

; Size:         162 code + 0 data bytes
; Run Time:       00:11.5
; Elapsed Time:   00:33.2
; Lines/CPU Min:   7158
; Lexemes/CPU-Min: 40302
; Memory Used:  146 pages
; Compilation Complete
```

FORREADIF LIS
FORREADIO LIS
FORRECPRO LIS
FORREWRSU LIS
FORSIGNAL LIS
FORREADKO LIS
FORREWIND LIS
FORREADSF LIS
FORREWRSO LIS
FORREADSN LIS
FORREADDU LIS
FORREADSU LIS
FORSECNDS LIS
FORREADIL LIS
FORREADKF LIS
FORREADKU LIS
FORREWRSF LIS
FORREADSL LIS
FORREADSO LIS