```
RRRRRRRRRRR      UUU          UUU  NNN         NNN     000000000     FFFFFFFFFFFFFFF   FFFFFFFFFFFFFFF
RRRRRRRRRRRR     UUU          UUU  NNN         NNN     000000000     FFFFFFFFFFFFFFF   FFFFFFFFFFFFFFF
RRRRRRRRRRRR     UUU          UUU  NNN         NNN     000000000     FFFFFFFFFFFFFFF   FFFFFFFFFFFFFFF
RRR          RRR UUU          UUU  NNN         NNN  000          000 FFF               FFF
RRR          RRR UUU          UUU  NNN         NNN  000          000 FFF               FFF
RRR          RRR UUU          UUU  NNN         NNN  000          000 FFF               FFF
RRR          RRR UUU          UUU  NNNNNN      NNN  000          000 FFF               FFF
RRR          RRR UUU          UUU  NNNNNN      NNN  000          000 FFF               FFF
RRR          RRR UUU          UUU  NNNNNN      NNN  000          000 FFF               FFF
RRRRRRRRRRR      UUU          UUU  NNN   NNN   NNN  000          000 FFFFFFFFFFF       FFFFFFFFFFF
RRRRRRRRRRRR     UUU          UUU  NNN   NNN   NNN  000          000 FFFFFFFFFFF       FFFFFFFFFFF
RRRRRRRRRRRR     UUU          UUU  NNN   NNN   NNN  000          000 FFFFFFFFFFF       FFFFFFFFFFF
RRR   RRR        UUU          UUU  NNN       NNNNNN 000          000 FFF               FFF
RRR   RRR        UUU          UUU  NNN       NNNNNN 000          000 FFF               FFF
RRR   RRR        UUU          UUU  NNN       NNNNNN 000          000 FFF               FFF
RRR      RRR     UUU          UUU  NNN          NNN 000          000 FFF               FFF
RRR      RRR     UUU          UUU  NNN          NNN 000          000 FFF               FFF
RRR      RRR     UUU          UUU  NNN          NNN 000          000 FFF               FFF
RRR         RRR  UUUUUUUUUUUUUUU  NNN          NNN    000000000     FFF               FFF
RRR         RRR  UUUUUUUUUUUUUUU  NNN          NNN    000000000     FFF               FFF
RRR         RRR  UUUUUUUUUUUUUUU  NNN          NNN    000000000     FFF               FFF
RRR         RRR  UUUUUUUUUUUUUUU  NNN          NNN    000000000     FFF               FFF
RRR         RRR  UUUUUUUUUUUUUUU  NNN          NNN    000000000     FFF               FFF
```

```
_$2

Sym
---
NDX
NDX
NUM
NUM
OPE
OUT
PAC
PAC
PAC
PAC
PAC
PAC
PAC
PAD
PAG
PAG
PAG
PAG
PAG
PAG
PAG
PER
PUT
RCO
RIN

RLI
RNO
RNO
RTY
SAV
STR
STR
STR
STR
STR
STR

STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
```

FILEID**GETLIN

```
  GGGGGGGG  EEEEEEEEEE  TTTTTTTTTT  LL            IIIIII    NN      NN
  GGGGGGGG  EEEEEEEEEE  TTTTTTTTTT  LL            IIIIII    NN      NN
GG          EE              TT      LL              II      NN      NN
GG          EE              TT      LL              II      NN      NN
GG          EE              TT      LL              II      NNNN    NN
GG          EE              TT      LL              II      NNNN    NN
GG          EEEEEEEE        TT      LL              II      NN  NN  NN
GG          EEEEEEEE        TT      LL              II      NN  NN  NN
GG  GGGGGG  EE              TT      LL              II      NN    NNNN
GG  GGGGGG  EE              TT      LL              II      NN    NNNN
GG      GG  EE              TT      LL              II      NN      NN      ....
GG      GG  EE              TT      LL              II      NN      NN      ....
  GGGGGG    EEEEEEEEEE      TT      LLLLLLLLLL    IIIIII    NN      NN      ....
  GGGGGG    EEEEEEEEEE      TT      LLLLLLLLLL    IIIIII    NN      NN      ....


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 getlin ( IDENT = 'V04-000'
;     2        P 0002  0                 %BLISS32[, ADDRESSING_MODE (EXTERNAL    = LONG_RELATIVE,
;     3          0003  0                                             NONEXTERNAL = LONG_RELATIVE)]
;     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:     DSR (Digital Standard RUNOFF) / DSRPLUS
;    34          0034  1 !
;    35          0035  1 ! ABSTRACT:     Picks up a line of text and centers it for output.
;    36          0036  1 !
;    37          0037  1 ! ENVIRONMENT:  Transportable
;    38          0038  1 !
;    39          0039  1 ! AUTHOR:       R.W.Friday       CREATION DATE: July, 1978
;    40          0040  1 !
;    41          0041  1
```

```
;    43          0042  1 %SBTTL 'Revision History'
;    44          0043  1 !
;    45          0044  1 ! MODIFIED BY:
;    46          0045  1 !
;    47          0046  1 !       005     KFA00005        Ken Alden       16-Mar-1983
;    48          0047  1 !               PUSH/POP_SCA now visible to DSR.
;    49          0048  1 !
;    50          0049  1 !       004     RER00004        Ron Randall     07-Mar-1983
;    51          0050  1 !               Global edit of all modules. Updated module names, idents,
;    52          0051  1 !               copyright dates. Changed require files tc BLISS library.
;    53          0052  1 !
;    54          0053  1 !--
;    55          0054  1
```

```
;    57          0055  1 %SBTTL 'Module Level Declarations'
;    58          0056  1
;    59          0057  1 !
;    60          0058  1 ! TABLE OF CONTENTS:
;    61          0059  1 !
;    62          0060  1 ! INCLUDE FILES:
;    63          0061  1 !
;    64          0062  1 LIBRARY 'NXPORT:XPORT';                 ! XPORT Library
;    65          0063  1 REQUIRE 'REQ:RNODEF';                   ! RUNOFF variant definitions
;    66          0194  1
;    67        U 0195  1 %IF DSRPLUS %THEN
;    68        U 0196  1 LIBRARY 'REQ:DPLLIB';                   ! DSRPLUS BLISS Library
;    69          0197  1 %ELSE
;    70          0198  1 LIBRARY 'REQ:DSRLIB';                   ! DSR BLISS Library
;    71          0199  1 %FI
;    72          0200  1
;    73          0201  1 !
;    74          0202  1 ! EXTERNAL REFERENCES:
;    75          0203  1 !
;    76          0204  1 EXTERNAL LITERAL
;    77          0205  1     RINTES : UNSIGNED (8);
;    78          0206  1
;    79          0207  1 EXTERNAL
;    80          0208  1     GCA : GCA_DEFINITION,
;    81          0209  1     MRA : REF FIXED_STRING,
;    82          0210  1     SCA : SCA_DEFINITION,
;    83          0211  1     TSF : TSF_DEFINITION,
;    84          0212  1     TTABLE : COUNTED_LIST;
;    85          0213  1
;    86          0214  1 EXTERNAL ROUTINE
;    87          0215  1     ENDCHR,
;    88          0216  1     ENDWRD,
;    89          0217  1     ERMA,
;    90          0218  1     ERML,
;    91          0219  1     OUTNJ,
;    92          0220  1     SCANT,
;    93          0221  1     SETCAS;
;    94          0222  1
;    95          0223  1 EXTERNAL LITERAL                        !Frror messages
;    96          0224  1     RNFCJL,
;    97          0225  1     RNFTTL;
;    98          0226  1
;    99          0227  1 !
;   100          0228  1 ! OWN STORAGE:
;   101          0229  1 !
;   102          0230  1 OWN
;   103          0231  1     PP_SCA : $H_R_SCA_BLOCK;    !Used in PUSH_SCA, POP_SCA macros (defined in SCA.REQ).
```

```
;  105     0232  1 GLOBAL ROUTINE GETLIN (PRESCAN, DO_CASE, TRANSLATION, DO_OUTPUT) : NOVALUE =
;  106     0233  1
;  107     0234  1 !++
;  108     0235  1 ! FUNCTIONAL DESCRIPTION:
;  109     0236  1 !
;  110     0237  1 !       Starting with the current position, GETLIN parses up to
;  111     0238  1 !       the first ';' or end-of-line; the resulting line is
;  112     0239  1 !       centered and output.
;  113     0240  1 !
;  114     0241  1 ! FORMAL PARAMETERS:
;  115     0242  1 !
;  116     0243  1 !       PRESCAN         - If true, scanning stops when the first ';'
;  117     0244  1 !                         is found.  Otherwise, scanning goes to the
;  118     0245  1 !                         end of the record.
;  119     0246  1 !       DO_CASE         - If true, then TRANSLATION indicates which
;  120     0247  1 !                         case rules are to be applied.
;  121     0248  1 !       TRANSLATION     -
;  122     0249  1 !       DO_OUTPUT       - If true, then output the scanned text,
;  123     0250  1 !                         else, skip output related code.
;  124     0251  1 !
;  125     0252  1 ! IMPLICIT INPUTS:      None
;  126     0253  1 !
;  127     0254  1 ! IMPLICIT OUTPUTS:     None
;  128     0255  1 !
;  129     0256  1 ! ROUTINE VALUE:
;  130     0257  1 ! COMPLETION CODES:     None
;  131     0258  1 !
;  132     0259  1 ! SIDE EFFECTS:         None
;  133     0260  1 !--
;  134     0261  1
;  135     0262  2     BEGIN
;  136     0263  2     LOCAL
;  137     0264  2         HOLD_TAB_COUNT,
;  138     0265  2         SCA_HOLD : VECTOR [SCA_SIZE];
;  139     0266  2
;  140     0267  2                 !Preserve SCA so flags in text don't cause changes
;  141     0268  2                 !to the formatting environment.
;  142     0269  2
;  143     0270  2     PUSH_SCA;   !Save the Special SAVEd SCA bits.
;  144     0271  2
;  145     0272  2     INCR I FROM 0 TO SCA_SIZE - 1 DO
;  146     0273  2         SCA_HOLD [.I] = .SCA [.I];
;  147     0274  2
;  148     0275  2         !Set up SCA so SCANT preserves white space; lots of room
;  149     0276  2         !is provided so a new line doesn't get started unless
;  150     0277  2         !the user makes an error.
;  151     0278  2     SCA_FILL = FALSE;
;  152     0279  2     SCA_JUSTIFY = FALSE;
;  153     0280  2     SCA_LM = 0;
;  154     0281  2     SCA_RM = 150;                               !Maximum width line.
;  155     0282  2     SCA_PRESCAN = .PRESCAN;                     !Indicate when processing stops.
;  156     0283  2
;  157     0284  2     IF  .DO_CASE
;  158     0285  2     THEN
;  159     0286  2         SETCAS (.TRANSLATION);                  !Establish specified case rules.
;  160     0287  2
;  161     0288  2         !Preserve tab count; temporarily set it to zero
```

```
;  162     0289  2          !so tabs get treated like spaces.
;  163     0290  2      HOLD_TAB_COUNT = .TTABLE [CL_INDEX];
;  164     0291  2      TTABLE [CL_INDEX] = 0;
;  165     0292  2
;  166     0293  2      SCANT ();    !Scan one input line.
;  167     0294  2
;  168     0295  2                   ! Drop trailing spaces, unless at least one of them is
;  169     0296  2                   ! underlined.  SCA_WRD_CPEND is equal to RINTES if a space/tab
;  170     0297  2                   ! was encountered after the last character on the line.
;  171     0298  2
;  172     0299  2      IF  .SCA_WRD_CPEND EQL RINTES
;  173     0300  2      THEN
;  174     0301  2
;  175     0302  2          IF  .SCA_WRD_LST_UND EQL 0
;  176     0303  2          THEN
;  177     0304  2              !None of the trailing spaces were underlined.
;  178     0305  2              !This means that it is safe to drop them.
;  179     0306  2              SCA_WRD_LST_SP = 0;
;  180     0307  2
;  181     0308  2      !The call on ENDWRD is made here, rather than letting
;  182     0309  2      !OUTNJ do it.  The reason it's done here is so that
;  183     0310  2      !TSF_EXT_HL gets updated, so that the length of the
;  184     0311  2      !text can be obtained.
;  185     0312  2      ENDWRD (FALSE, FALSE, FALSE);
;  186     0313  2
;  187     0314  2      !Check to see that the text retrieved is not too long
;  188     0315  2      IF  .TSF_EXT_HL GTR .GCA_LWIDTH
;  189     0316  2      THEN
;  190     0317  3          BEGIN
;  191     0318  3          ERMA (RNFTTL, FALSE);   !Text cannot possibly be centered
;  192     0319  3          TSF_EXT_HL = 0; !Fix up TSF_EXT_HL. The entire line still gets printed.
;  193     0320  2          END;
;  194     0321  2
;  195     0322  2      !Restore SCA so actual margins can be used.
;  196     0323  2      INCR I FROM SCA_CASE_SIZE TO SCA_SIZE - 1 DO        !NOTE: modified case rules "play through".
;  197     0324  2          SCA [.I] = .SCA_HOLD [.I];
;  198     0325  2
;  199     0326  2      POP_SCA;     !Restore the special SAVEd SCA bits.
;  200     0327  2
;  201     0328  2      TTABLE [CL_INDEX] = .HOLD_TAB_COUNT;                !Restore tab count
;  202     0329  2
;  203     0330  2      IF  NOT .DO_OUTPUT  ! If output was not wanted,
;  204     0331  2          THEN RETURN;    !   it best we return now.  Otherwise, proceding
;  205     0332  2                          !   beyond here will generate output.
;  206     0333  2
;  207     0334  2      !Compute and validate number of spaces that have to
;  208     0335  2      !be inserted on the left to center the line
;  209     0336  2      TSF_ADJUST = .SCA_LM + (.SCA_RM - .SCA_LM - .TSF_EXT_HL)/2;
;  210     0337  2
;  211     0338  2      IF  .TSF_ADJUST LSS 0                                                    ; F
;  212     0339  2      THEN
;  213     0340  3          BEGIN
;  214     0341  3          ERML (RNFCJL);  !It's not possible to center the line.               ;
;  215     0342  3          TSF_ADJUST = 0; !Setting tsf_adjust to zero causes the line
;  216     0343  2          END;            !to go out against the left edge of the page.
;  217     0344  2
;  218     0345  2      !Force out the line of text
```

```
;  219          0346  2     OUTNJ ();
;  220          0347  1     END;                    !End of GETLIN


                                                    .TITLE  GETLIN
                                                    .IDENT  \V04-000\

                                                    .PSECT  $OWN$,NOEXE,2

                                          00000 PP_SCA: .BLKB   48

                                                    .EXTRN  RINTES, GCA, MRA
                                                    .EXTRN  SCA, TSF, TTABLE
                                                    .EXTRN  ENDCHR, ENDWRD, ERMA
                                                    .EXTRN  ERML, OUTNJ, SCANT
                                                    .EXTRN  SETCAS, RNFCJL, RNFTTL

                                                    .PSECT  $CODE$,NOWRT,2

                                     007C 00000     .ENTRY  GETLIN, Save R2,R3,R4,R5,R6                 ; 0232
                          56 00000000G   EF  9E 00002   MOVAB   TTABLE+4, R6
                          55 00000000G   EF  9E 00009   MOVAB   TSF, R5
                          54 00000000'   EF  9E 00010   MOVAB   PP_SCA, R4
                          53 00000000G   EF  9E 00017   MOVAB   SCA+116, R3
                          5E     FE80    CE  9E 0001E   MOVAB   -384(SP), SP
                          64       F0    B3  D0 00023   MOVL    @SCA+100, PP_SCA                      ; 0265
                    04    A4       F4    B3  D0 00027   MOVL    @SCA+104, PP_SCA+4
                    08    A4       F8    B3  D0 0002C   MOVL    @SCA+108, PP_SCA+8
                    0C    A4       FC    B3  D0 00031   MOVL    @SCA+112, PP_SCA+12
                    10    A4       00    B3  D0 00036   MOVL    @SCA+116, PP_SCA+16
                    14    A4       04    B3  D0 0003B   MOVL    @SCA+120, PP_SCA+20
                    18    A4       08    B3  D0 00040   MOVL    @SCA+124, PP_SCA+24
                    1C    A4       0C    B3  D0 00045   MOVL    @SCA+128, PP_SCA+28
                    20    A4       10    B3  D0 0004A   MOVL    @SCA+132, PP_SCA+32
                    24    A4       14    B3  D0 0004F   MOVL    @SCA+136, PP_SCA+36
                    28    A4       18    B3  D0 00054   MOVL    @SCA+140, PP_SCA+40
                    2C    A4       1C    B3  D0 00059   MOVL    @SCA+144, PP_SCA+44
                                     50    D4 0005E       CLRL    I                                    ; 0272
                        6E40         8C A340  D0 00060 1$: MOVL    SCA[I], SCA_HOLD[I]                  ; 0273
                F2        50 0000005F    8F  F3 00066       AOBLEQ  #95, I, 1$
                                         F4  B3  D4 0006E   CLRL    @SCA+104                             ; 0278
                                         F0  B3  D4 00071   CLRL    @SCA+100                             ; 0279
                                         00  B3  D4 00074   CLRL    @SCA+116                             ; 0280
                          04    B3       96  8F  9A 00077   MOVZBL  #150, @SCA+120                       ; 0281
                          38    A3       04  AC  D0 0007C   MOVL    PRESCAN, SCA+172                     ; 0282
                                0A       08  AC  E9 00081   BLBC    DO_CASE, 2$                          ; 0284
                                         0C  AC  DD 00085   PUSHL   TRANSLATION                          ; 0286
                000000000G      EF           01  FB 00088   CALLS   #1, SETCAS
                                52           66  D0 0008F 2$: MOVL    TTABLE+4, HOLD_TAB_COUNT             ; 0290
                                             66  D4 00092   CLRL    TTABLE+4                             ; 0291
                000000000G      EF           00  FB 00094   CALLS   #0, SCANT                            ; 0293
                000000000G      8F      00A4 C3  D1 0009B   CMPL    SCA+280, #RINTES                     ; 0299
                                             0A  12 000A4   BNEQ    3$
                                        00E0 C3  D5 000A6   TSTL    SCA+340                              ; 0302
                                             04  12 000AA   BNEQ    3$
                                        00D8 C3  D4 000AC   CLRL    SCA+332                              ; 0306
                                             7E  7C 000B0 3$: CLRQ    -(SP)                                ; 0312
```

```
                                      7E  D4 000B2        CLRL    -(SP)                                ;
              00000000G  EF           03  FB 000B4        CALLS   #3, ENDWRD                           ;
                         50           65  D0 000BB        MOVL    TSF, R0                              ; 0315
              00000000G  FF       04  A0  D1 000BE        CMPL    4(R0), @GCA+140                      ;
                                      15  15 000C6        BLEQ    4$                                   ;
                                      7E  D4 000C8        CLRL    -(SP)                                ; 0318
                           00000000G  8F  DD 000CA        PUSHL   #RNFTTL                              ;
              00000000G  EF           02  FB 000D0        CALLS   #2, ERMA                             ;
                         50           65  D0 000D7        MOVL    TSF, R0                              ;
                                  04  A0  D4 000DA        CLRL    4(R0)                                ; 0319
                         50           19  D0 000DD 4$:    MOVL    #25, I                               ; 0323
                   8C  A340         6E40  D0 000E0 5$:    MOVL    SCA_HOLD[I], SCA[I]                  ; 0324
          F2             50  0000005F  8F  F3 000E6        AOBLEQ  #95, I, 5$                           ;
                   F0    B3           64  D0 000EE        MOVL    PP_SCA, @SCA+100                     ;
                   F4    B3       04  A4  D0 000F2        MOVL    PP_SCA+4, @SCA+104                   ;
                   F8    B3       08  A4  D0 000F7        MOVL    PP_SCA+8, @SCA+108                   ;
                   FC    B3       0C  A4  D0 000FC        MOVL    PP_SCA+12, @SCA+112                  ;
                   00    B3       10  A4  D0 00101        MOVL    PP_SCA+16, @SCA+116                  ;
                   04    B3       14  A4  D0 00106        MOVL    PP_SCA+20, @SCA+120                  ;
                   08    B3       18  A4  D0 0010B        MOVL    PP_SCA+24, @SCA+124                  ;
                   0C    B3       1C  A4  D0 00110        MOVL    PP_SCA+28, @SCA+128                  ;
                   10    B3       20  A4  D0 00115        MOVL    PP_SCA+32, @SCA+132                  ;
                   14    B3       24  A4  D0 0011A        MOVL    PP_SCA+36, @SCA+136                  ;
                   18    B3       28  A4  D0 0011F        MOVL    PP_SCA+40, @SCA+140                  ;
                   1C    B3       2C  A4  D0 00124        MOVL    PP_SCA+44, @SCA+144                  ;
                         66           52  D0 00129        MOVL    HOLD_TAB_COUNT, TTABLE+4             ; 0328
                         32       10  AC  E9 0012C        BLBC    DO_OUTPUT, 7$                        ; 0330
                         50           65  D0 00130        MOVL    TSF, R0                              ; 0331
          51       04    B3       00  B3  C3 00133        SUBL3   @SCA+116, @SCA+120, R1               ; 0336
                         51       04  A0  C2 00139        SUBL2   4(R0), R1                            ;
                         51           02  C6 0013D        DIVL2   #2, R1                               ;
     28   A0             51       00  B3  C1 00140        ADDL3   @SCA+116, R1, 40(R0)                 ;
                                      13  18 00146        BGEQ    6$                                   ; 0338
                           00000000G  8F  DD 00148        PUSHL   #RNFCJL                              ; 0341
              00000000G  EF           01  FB 0014E        CALLS   #1, ERML                             ;
                         50           65  D0 00155        MOVL    TSF, R0                              ;
                                  28  A0  D4 00158        CLRL    40(R0)                               ; 0342
              00000000G  EF           00  FB 0015B 6$:    CALLS   #0, OUTNJ                            ; 0346
                                          04 00162 7$:    RET                                          ; 0347

; Routine Size:  355 bytes,    Routine Base:  $CODE$ + 0000


;  221          0348  1
;  222          0349  1 END                               !End of module
;  223          0350  0 ELUDOM



;                                 PSECT SUMMARY
;
;       Name                   Bytes                        Attributes
;
;  $OWN$                          48  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
```

```
;  $CODE$                          355  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)



;                              Library Statistics
;
;                                        -------- Symbols --------      Pages       Processing
;       File                             Total    Loaded   Percent      Mapped      Time
;
;  _$255$DUA28:[SYSLIB]XPORT.L32;1         590         0         0         252       00:00.2
;  _$255$DUA28:[RUNOFF.SRC]DSRLIB.L32;1   1248        47         3          86       00:00.3




;                                    COMMAND QUALIFIERS

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

; Size:         355 code + 48 data bytes
; Run Time:         00:09.6
; Elapsed Time:     00:28.8
; Lines/CPU Min:     2185
; Lexemes/CPU-Min: 18293
; Memory Used:  95 pages
; Compilation Complete
```