```
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
```

```
  CCCCCCCC    AAAAAA    RRRRRRRR      AAAAAA      GGGGGGGG  EEEEEEEEEE
  CCCCCCCC    AAAAAA    RRRRRRRR      AAAAAA      GGGGGGGG  EEEEEEEEEE
CC           AA    AA   RR     RR   AA    AA   GG          EE
CC           AA    AA   RR     RR   AA    AA   GG          EE
CC           AA    AA   RR     RR   AA    AA   GG          EE
CC           AA    AA   RR     RR   AA    AA   GG          EE
CC           AA    AA   RRRRRRRR    AA    AA   GG          EEEEEEEE
CC           AA    AA   RRRRRRRR    AA    AA   GG          EEEEEEEE
CC           AAAAAAAAAA   RR   RR     AAAAAAAAAA   GG   GGGGGG  EE
CC           AAAAAAAAAA   RR   RR     AAAAAAAAAA   GG   GGGGGG  EE
CC           AA    AA   RR     RR   AA    AA   GG      GG  EE              ....
CC           AA    AA   RR     RR   AA    AA   GG      GG  EE              ....
  CCCCCCCC   AA    AA   RR      RR  AA    AA     GGGGGG    EEEEEEEEEE      ....
  CCCCCCCC   AA    AA   RR      RR  AA    AA     GGGGGG    EEEEEEEEEE      ....


LL           IIIIII      SSSSSSSS
LL           IIIIII      SSSSSSSS
LL             II      SS
LL             II      SS
LL             II      SS
LL             II      SS
LL             II        SSSSSS
LL             II        SSSSSS
LL             II              SS
LL             II              SS
LL             II              SS
LL             II              SS
LLLLLLLLLL   IIIIII    SSSSSSSS
LLLLLLLLLL   IIIIII    SSSSSSSS
```

```
;     1          0001  0 MODULE CARAGE ( 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 !*  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:     DSR (Digital Standard RUNOFF) / DSRPLUS
;    33          0033  1 !
;    34          0034  1 ! ABSTRACT:
;    35          0035  1 !
;    36          0036  1 !       Processes the .PAGE, .SKIP, .BLANK, .TEST PAGE,
;    37          0037  1 !       .BREAK, and .SPACING commands.
;    38          0038  1 !
;    39          0039  1 ! ENVIRONMENT:  Transportable
;    40          0040  1 !
;    41          0041  1 ! AUTHOR:       R.W.Friday      CREATION DATE:  May, 1978
;    42          0042  1
```

```
;    44          0043  1 %SBTTL 'Revision History'
;    45          0044  1 ! MODIFIED BY:
;    46          0045  1 !
;    47          0046  1 !       006     KFA00006        Ken Alden       05-Aug-1983
;    48          0047  1 !               Added handling for .PAGE NOW (plus).
;    49          0048  1 !
;    50          0049  1 !       005     RER00005        Ron Randall     17-Mar-1983
;    51          0050  1 !               For DSRPLUS:
;    52          0051  1 !               In [h_skip], added code to test for negative skip in topnote.
;    53          0052  1 !
;    54          0053  1 !       004     KFA00004        Ken Alden       07-Mar-1983
;    55          0054  1 !               Global edit of all modules. Updated module names, idents,
;    56          0055  1 !               copyright dates. Changed require files to BLISS library.
;    57          0056  1 !--
;    58          0057  1
```

```
;    60          0058  1 %SBTTL 'Module Level Declarations'
;    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
;    77          0205  1     FNCT        : FNCT_DEFINITION,
;    78          0206  1     IRA         : FIXED_STRING,
;    79          0207  1     MRA         : REF FIXED_STRING,
;    80          0208  1     NUMPRM      : NUMPRM_DEFINE,
;    81          0209  1     PHAN        : PHAN_DEFINITION,
;    82          0210  1     SCA         : SCA_DEFINITION;
;    83          0211  1
;    84        U 0212  1 %IF DSRPLUS %THEN
;    85        U 0213  1 EXTERNAL
;    86        U 0214  1     TOPNOT      : TN_DEFINITION;
;    87        U 0215  1
;    88        U 0216  1 EXTERNAL LITERAL
;    89        U 0217  1     RNFIIT;
;    90          0218  1 %FI
;    91          0219  1
;    92          0220  1 EXTERNAL LITERAL
;    93          0221  1     RNFIFT,
;    94          0222  1     RNFINM,
;    95          0223  1     RNFMNL;
;    96          0224  1
;    97          0225  1 EXTERNAL ROUTINE
;    98          0226  1     erma,
;    99          0227  1     gcpage,
;   100          0228  1     gcpos,
;   101          0229  1     gcskip,
;   102          0230  1     gtpc,
;   103        U 0231  1 %IF DSRPLUS %THEN
;   104        U 0232  1     newpag,
;   105          0233  1 %FI
;   106          0234  1     tpffit,
;   107          0235  1     tpfsiz;
;   108          0236  1
```

```
;  110          0237  1 GLOBAL ROUTINE carage (handler_code) : NOVALUE =
;  111          0238  1
;  112          0239  1 !++
;  113          0240  1 ! FUNCTIONAL DESCRIPTION:
;  114          0241  1 !
;  115          0242  1 !       See the ABSTRACT, above.
;  116          0243  1 !
;  117          0244  1 ! FORMAL PARAMETERS:
;  118          0245  1 !
;  119          0246  1 !       handler_code - Indicates which command is to be processed.
;  120          0247  1 !
;  121          0248  1 ! IMPLICIT INPUTS:
;  122          0249  1 !
;  123          0250  1 !       numprm - Expected to contain a number, as processed by GETNUM.
;  124          0251  1 !
;  125          0252  1 ! IMPLICIT OUTPUTS:     None
;  126          0253  1 !
;  127          0254  1 ! ROUTINE VALUE:
;  128          0255  1 ! COMPLETION CODES:     None
;  129          0256  1 !
;  130          0257  1 ! SIDE EFFECTS:         None
;  131          0258  1 !--
;  132          0259  1
;  133          0260  2     BEGIN
;  134          0261  2
;  135          0262  2     ! Some commands take a number.
;  136          0263  2     IF  (.HANDLER_CODE NEQ H_BREAK) AND
;  137          0264  3         (.HANDLER_CODE NEQ H_PAGE)
;  138        U 0265  3 %IF DSRPLUS %THEN
;  139        U 0266  3                 AND
;  140        U 0267  3         (.HANDLER_CODE NEQ H_PAGE_NOW)
;  141          0268  3 %FI
;  142          0269  2     THEN
;  143          0270  2
;  144          0271  2         IF  NOT .NUM_RESULT
;  145          0272  2         ! Ignore command if number was bad.
;  146          0273  2         THEN
;  147          0274  2             RETURN;
;  148          0275  2
;  149          0276  2     SELECTONE .HANDLER_CODE OF
;  150          0277  2         SET
;  151          0278  2
;  152          0279  2         [H_BLANK, H_SKIP] :
;  153          0280  3             BEGIN
;  154          0281  3
;  155          0282  3             IF  .NUM_LENGTH EQL 0
;  156          0283  3             THEN
;  157          0284  3                 NUM_VALUE = 1;                  ! Default if no parameter.
;  158          0285  3
;  159          0286  3             IF  .HANDLER_CODE EQL H_SKIP
;  160          0287  3             ! Commands .SKIP and .BLANK mean the same
;  161          0288  3             ! thing, except that .SKIP means ".SPACING"
;  162          0289  3             ! blank lines, whereas .BLANK means exactly
;  163          0290  3             ! as many blank lines as specified.
;  164          0291  3             THEN
;  165          0292  3                 NUM_VALUE = .NUM_VALUE * .SCA_SPACING;
;  166          0293  3
```

```
;  167          0294  3              IF  .NUM_VALUE LSS 0
;  168          0295  3              THEN
;  169          0296  4                  BEGIN
;  170          0297  4
;  171        U 0298  4 %IF DSRPLUS %THEN
;  172        U 0299  4                  !
;  173        U 0300  4                  ! .SKIP with negative value is illegal in a topnote.
;  174        U 0301  4                  !
;  175        U 0302  4                  IF  .TN_COLLECTING
;  176        U 0303  4                  THEN
;  177        U 0304  4                      BEGIN
;  178        U 0305  4                      ERMA (RNFIIT, FALSE);
;  179        U 0306  4                      RETURN;
;  180        U 0307  4                      END;
;  181          0308  4 %FI
;  182          0309  4
;  183          0310  4                  !
;  184          0311  4                  ! .SKIP with negative value is illegal in a footnote.
;  185          0312  4                  !
;  186          0313  4                  IF  .FNCT_COLLECTING
;  187          0314  4                  THEN
;  188          0315  5                      BEGIN
;  189          0316  5                      ERMA (RNFIFT, FALSE);
;  190          0317  5                      RETURN;
;  191          0318  4                      END;
;  192          0319  4
;  193          0320  4                  IF  ABS (.NUM_VALUE) GEQ .PHAN_LLINES
;  194          0321  4                  THEN
;  195          0322  4                      ! User said something like .SKIP -9999.
;  196          0323  5                      BEGIN
;  197          0324  5                      ERMA (RNFINM, FALSE);
;  198          0325  5                      RETURN;
;  199          0326  4                      END;
;  200          0327  4
;  201          0328  4                  ! It's a valid .SKIP/.BLANK with a negative count.
;  202          0329  4                  GCPOS (.NUM_VALUE);
;  203          0330  4                  RETURN;
;  204          0331  3                  END;
;  205          0332  3
;  206          0333  3              ! Generate code for the conditional skip.
;  207          0334  3              GCSKIP (.NUM_VALUE);
;  208          0335  2              END;
;  209          0336  2
;  210          0337  2          [H_BREAK] :
;  211          0338  2              ! Everything associated with .BREAK is contained in
;  212          0339  2              ! the tables, that are interpreted by DOCM.
;  213          0340  2              0;
;  214          0341  2
;  215          0342  2          [H_PAGE] :
;  216          0343  2              ! Generate code for the conditional page.
;  217          0344  2              GCPAGE ();
;  218          0345  2
;  219        U 0346  2 %IF DSRPLUS %THEN
;  220        U 0347  2          [H_PAGE_NOW] :
;  221        U 0348  2              ! Throw   page NOW.
;  222        U 0349  2              BEGIN
;  223        U 0350  2              IF  .fnct_collecting                ! Don't start a new page if
```

```
;  224     U 0351  2              THEN                                    ! footnotes are being collected,
;  225     U 0352  2                  RETURN;                             ! since footnotes all belong on
;  226     U 0353  2                                                      ! one page.
;  227     U 0354  2
;  228     U 0355  2              phan_top_page = false;                  ! Start another one, even
;  229     U 0356  2              newpag ();                              ! if we're at the beginning of a
;  230     U 0357  2              END;                                    ! new page.
;  231       0358  2 %FI
;  232       0359  2
;  233       0360  2          [H_SPACING] :
;  234       0361  3              BEGIN
;  235       0362  3
;  236       0363  3              IF  .NUM_LENGTH EQL 0
;  237       0364  3              ! No number provided.
;  238       0365  3              THEN
;  239       0366  4                  BEGIN
;  240       0367  4                  ERMA (RNFMNL, FALSE);
;  241       0368  4                  RETURN;
;  242       0369  3                  END;
;  243       0370  3
;  244       0371  3              IF  (.NUM_VALUE LEQ 0) OR
;  245       0372  4                  (.NUM_VALUE GTR 5)
;  246       0373  4              ! Number not between 0 and 5.
;  247       0374  3              THEN
;  248       0375  4                  BEGIN
;  249       0376  4                  ERMA (RNFINM, FALSE);
;  250       0377  4                  RETURN;
;  251       0378  3                  END;
;  252       0379  3
;  253       0380  3              SCA_SPACING = .NUM_VALUE;
;  254       0381  2              END;
;  255       0382  2
;  256       0383  2          [H_TEST_PAGE] :
;  257       0384  3              BEGIN
;  258       0385  3
;  259       0386  3              IF  .NUM_LENGTH EQL 0
;  260       0387  3              ! No number provided.
;  261       0388  3              THEN
;  262       0389  4                  BEGIN
;  263       0390  4                  ERMA (RNFMNL, FALSE);
;  264       0391  4                  RETURN;
;  265       0392  3                  END;
;  266       0393  3
;  267       0394  3              IF  .NUM_VALUE LEQ 0
;  268       0395  3              ! Negative number not allowed.
;  269       0396  3              THEN
;  270       0397  4                  BEGIN
;  271       0398  4                  ERMA (RNFINM, FALSE);
;  272       0399  4                  RETURN;
;  273       0400  3                  END;
;  274       0401  3
;  275       0402  3              ! Add count of pending footnote lines, if any,
;  276       0403  3              ! to the given test-page value.
;  277       0404  3              ! Generate code for a test page.
;  278       0405  3              GTPC (.NUM_VALUE + TPFSIZ (TPFFIT ()));
;  279       0406  2              END;
;  280       0407  2
```

```
;  281          0408  2          TES;
;  282          0409  2
;  283          0410  1      END;                                    !End of CARAGE


                                                                     .TITLE  CARAGE
                                                                     .IDENT  \V04-000\

                                                                     .EXTRN  FNCT, IRA, MRA, NUMPRM
                                                                     .EXTRN  PHAN, SCA, RNFIFT
                                                                     .EXTRN  RNFINM, RNFMNL, ERMA
                                                                     .EXTRN  GCPAGE, GCPOS, GCSKIP
                                                                     .EXTRN  GTPC, TPFFIT, TPFSIZ

                                                                     .PSECT  $CODE$,NOWRT,2

                                                 000C 00000          .ENTRY  CARAGE, Save R2,R3                  ; 0237
                                53 00000000G   EF   9E 00002         MOVAB   NUMPRM+4, R3
                                52       04    AC   D0 00009         MOVL    HANDLER_CODE, R2                    ; 0263
                                08             52   D1 0000D         CMPL    R2, #11
                                               0E   13 00010         BEQL    1$
                          000000A7  8F         52   D1 00012         CMPL    R2, #167                            ; 0264
                                               05   13 00019         BEQL    1$
                                01       FC    A3   E8 0001B         BLBS    NUMPRM, 1$                          ; 0271
                                               04   0001F           RET
                                0A             52   D1 00020 1$:     CMPL    R2, #10                             ; 0279
                                               09   13 00023         BEQL    2$
                          000000BC  8F         52   D1 00025         CMPL    R2, #188
                                               52   12 0002C         BNEQ    8$
                                         08    A3   D5 0002E 2$:     TSTL    NUMPRM+12                           ; 0282
                                               03   12 00031         BNEQ    3$
                                63             01   D0 00033         MOVL    #1, NUMPRM+4                        ; 0284
                          000000BC  8F         52   D1 00036 3$:     CMPL    R2, #188                            ; 0286
                                               07   12 0003D         BNEQ    4$
                                63 00000000G   FF   C4 0003F         MULL2   @SCA+124, NUMPRM+4                  ; 0292
                                               63   D5 00046 4$:     TSTL    NUMPRM+4                            ; 0294
                                               2C   18 00048         BGEQ    7$
                                0A 00000000G   EF   E9 0004A         BLBC    FNCT+20, 5$                         ; 0313
                                               7E   D4 00051         CLRL    -(SP)                               ; 0316
                                   00000000G   8F   DD 00053         PUSHL   #RNFIFT
                                               7F   11 00059         BRB     14$
                                50             63   D0 0005B 5$:     MOVL    NUMPRM+4, R0                        ; 0320
                                               03   18 0005E         BGEQ    6$
                                50             50   CE 00060         MNEGL   R0, R0
                          00000000C  ??        50   D1 00063 6$:     CMPL    R0, @PHAN+4
                                               66   18 0006A         BGEQ    13$
                                               63   DD 0006C         PUSHL   NUMPRM+4                            ; 0329
                          00000000G  EF        01   FB 0006E         CALLS   #1, GCPOS
                                               04   00075           RET                                         ; 0296
                                               63   DD 00076 7$:     PUSHL   NUMPRM+4                            ; 0334
                          00000000G  EF        01   FB 00078         CALLS   #1, GCSKIP
                                               04   0007F           RET                                         ; 0276
                                0B             52   D1 00080 8$:     CMPL    R2, #11                             ; 0337
                                               78   13 00083         BEQL    16$
                          000000A7  8F         52   D1 00085         CMPL    R2, #167                            ; 0342
                                               08   12 0008C         BNEQ    9$
                          00000000G  EF        00   FB 0008E         CALLS   #0, GCPAGE                          ; 0344
```

```
                                                 04 00095          RET
                 000000BD   8F                  52 D1 00096 9$:   CMPL    R2, #189                    ;  0360
                                                 21 12 0009D       BNEQ    12$
                                           08    A3 D5 0009F       TSTL    NUMPRM+12                   ;  0363
                                                 0A 12 000A2       BNEQ    11$
                                                 7E D4 000A4 10$:  CLRL    -(SP)                       ;  0367
                                      00000000G  8F DD 000A6       PUSHL   #RNFMNL
                                                 2C 11 000AC       BRB     14$
                                      50         63 D0 000AE 11$:  MOVL    NUMPRM+4, R0                ;  0371
                                                 1F 15 000B1       BLEQ    13$
                                      05         50 D1 000B3       CMPL    R0, #5                      ;  0372
                                                 1A 14 000B6       BGTR    13$
                 00000000G  FF                   63 D0 000B8       MOVL    NUMPRM+4, @SCA+124          ;  0380
                                                 04 000BF          RET                                 ;  0276
                 000000CB   8F                   52 D1 000C0 12$:  CMPL    R2, #203                    ;  0383
                                                 34 12 000C7       BNEQ    16$
                                           08    A3 D5 000C9       TSTL    NUMPRM+12                   ;  0386
                                                 D6 13 000CC       BEQL    10$
                                                 63 D5 000CE       TSTL    NUMPRM+4                    ;  0394
                                                 10 14 000D0       BGTR    15$
                                                 7E D4 000D2 13$:  CLRL    -(SP)                       ;  0398
                                      00000000G  8F DD 000D4       PUSHL   #RNFINM
                 00000000G  EF                   02 FB 000DA 14$:  CALLS   #2, ERMA                    ;  0397
                                                 04 000E1          RET
                 00000000G  EF                   00 FB 000E2 15$:  CALLS   #0, TPFFIT                  ;  0405
                                                 50 DD 000E9       PUSHL   R0
                 00000000G  EF                   01 FB 000EB       CALLS   #1, TPFSIZ
                                           00 B340 9F 000F2        PUSHAB  @NUMPRM+4[R0]
                 00000000G  EF                   01 FB 000F6       CALLS   #1, GTPC
                                                 04 000FD 16$:     RET                                 ;  0410

; Routine Size:  254 bytes,    Routine Base:  $CODE$ + 0000

;  284          0411  1
;  285          0412  1 END                                        !End of module
;  286          0413  0 ELUDOM



;                                    PSECT SUMMARY
;
;      Name                      Bytes                          Attributes
;
;  $CODE$                          254  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        00:00.1
;  _$255$DUA28:[RUNOFF.SRC]DSRLIB.L32;1        1248       20        1       86        00:00.2



;                                       COMMAND QUALIFIERS

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

; Size:         254 code + 0 data bytes
; Run Time:         00:06.0
; Elapsed Time:     00:16.5
; Lines/CPU Min:     4123
; Lexemes/CPU-Min: 13617
; Memory Used:  73 pages
; Compilation Complete
```

BARS
LIS

SETCAS
REQ

VR
REQ

ALINE
LIS

CARAGE
LIS

SAVSTK
REQ

AREC
LIS

TOCRTY
REQ

XTNTAB
REQ

CAPTION
LIS

CLH
LIS

SCA
REQ

BOTPAG
LIS

TPROB
REQ

CNTVMSREQ
R32

TSF
REQ

NDXVMSREQ
R32

SAVE
REQ

CENTXT
LIS