```
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
RRRRRRRRRRRR     UUU          UUU  NNN  NNN     NNN  000         000  FFFFFFFFFFFF     FFFFFFFFFFFF
RRRRRRRRRRRR     UUU          UUU  NNN  NNN     NNN  000         000  FFFFFFFFFFFF     FFFFFFFFFFFF
RRRRRRRRRRRR     UUU          UUU  NNN  NNN     NNN  000         000  FFFFFFFFFFFF     FFFFFFFFFFFF
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
STR
STR
```

```
RRRRRRRR    TTTTTTTTTT  EEEEEEEEEE  RRRRRRRR    MM      MM
RRRRRRRR    TTTTTTTTTT  EEEEEEEEEE  RRRRRRRR    MM      MM
RR      RR      TT      EE          RR      RR  MMMM  MMMM
RR      RR      TT      EE          RR      RR  MMMM  MMMM
RR      RR      TT      EE          RR      RR  MM  MM  MM
RR      RR      TT      EE          RR      RR  MM  MM  MM
RRRRRRRR        TT      EEEEEEEE    RRRRRRRR    MM      MM
RRRRRRRR        TT      EEEEEEEE    RRRRRRRR    MM      MM
RR  RR          TT      EE          RR  RR      MM      MM
RR  RR          TT      EE          RR  RR      MM      MM
RR    RR        TT      EE          RR    RR    MM      MM      ....
RR    RR        TT      EE          RR    RR    MM      MM      ....
RR      RR      TT      EEEEEEEEEE  RR      RR  MM      MM      ....
RR      RR      TT      EEEEEEEEEE  RR      RR  MM      MM      ....

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
```

RTE
V04

```
;      1          0001  0 %TITLE 'RUNOFF termination and cleanup'
;      2          0002  0 MODULE rterm (  IDENT = 'V04-000'
;      3        P 0003  0                 %BLISS32 [, ADDRESSING_MODE (EXTERNAL    = LONG_RELATIVE,
;      4          0004  0                                              NONEXTERNAL = LONG_RELATIVE)]
;      5          0005  0                 ) =
;      6          0006  1 BEGIN
;      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:     Routine for normal termination of a document.
;     36          0036  1 !
;     37          0037  1 ! ENVIRONMENT:  Transportable
;     38          0038  1 !
;     39          0039  1 ! AUTHOR:       R.W.Friday      CREATION DATE:  April, 1978
;     40          0040  1
```

```
;    42          0041  1 %SBTTL 'Revision History'
;    43          0042  1 ! MODIFIED BY:
;    44          0043  1 !
;    45          0044  1 !       032     KFA00032        Ken Alden       21-Oct-1983
;    46          0045  1 !               RNFIPF would still get output if the last page of the file
;    47          0046  1 !                 was the one selected by the /PAGES qualifier. This has
;    48          0047  1 !                 been corrected.
;    49          0048  1 !
;    50          0049  1 !       031     KFA00031        Ken Alden       06-Oct-1983
;    51          0050  1 !               Added a warning message for users who said /PAGES but
;    52          0051  1 !                 issued a page string that never matched any pages in the doc.
;    53          0052  1 !
;    54          0053  1 !       030     REM00030        Ray Marshall    10-Aug-1983
;    55          0054  1 !               KWTRPN and OUTCREF were declared (but not used) when DSRPLUS
;    56          0055  1 !                 was false.  This caused LINKer errors when building RUNOFF.
;    57          0056  1 !
;    58          0057  1 !       029     REM00029        Ray Marshall    22-Jul-1983
;    59          0058  1 !               Defined TTYRES and DISPSI as NOVALUE to match the way they
;    60          0059  1 !                 are coded.
;    61          0060  1 !               Added BLISS36 ONLY logic at the end of routine RTERM to release
;    62          0061  1 !                 the dynamic memory used for the cross-ref data structures
;    63          0062  1 !                 back to the system. Then, we can clear the listhead (pointer)
;    64          0063  1 !                 so that when the .BRN is read on another run, it doesn't
;    65          0064  1 !                 have the old names lying around.  This was needed because
;    66          0065  1 !                 in this environment, the image does not have to be reloaded
;    67          0066  1 !                 for successive runs through DSR.  This is only a DSRPLUS
;    68          0067  1 !                 addition.
;    69          0068  1 !
;    70          0069  1 !       028     KFA00028        Ken Alden       19-Jul-1983
;    71          0070  1 !               Fixed the /pages bug with gca_page_cnt.
;    72          0071  1 !
;    73          0072  1 !       027     KFA00027        Ken Alden       14-Jul-1983
;    74          0073  1 !               Suppresed CRU if /auto.
;    75          0074  1 !               Removed CR3.
;    76          0075  1 !
;    77          0076  1 !       026     KFA00026        Ken Alden       13-Jul-1983
;    78          0077  1 !               Altered the output of cref info messages slightly.
;    79          0078  1 !
;    80          0079  1 !       025     KAD00025        Keith Dawson    10-Jun-1983
;    81          0080  1 !               Cref-changed condition slould not increment /AUTO count in
;    82          0081  1 !                 first pass (already been incremented).
;    83          0082  1 !
;    84          0083  1 !       024     RER00024        Ron Randall     1-Jun-1983
;    85          0084  1 !               Added code that checks for need to output deferred figures.
;    86          0085  1 !
;    87          0086  1 !       023     KAD00023        Keith Dawson    27-May-1983
;    88          0087  1 !               Make sure Cref-changed condition increments /AUTO count.
;    89          0088  1 !
;    90          0089  1 !       022     KAD00022        Keith Dawson    3-May-1983
;    91          0090  1 !               Suppress 'Cref-changed' messages on early /AUTO passes.
;    92          0091  1 !
;    93          0092  1 !       021     KAD00021        Keith Dawson    11-Apr-1983
;    94          0093  1 !               Add new termination messages (RNFPC2, PC3, PC4, PC5) for
;    95          0094  1 !                 information written to the .BRN file (and for DSRPLUS
;    96          0095  1 !                 cross-referencing, also read from the input .BRN file).
;    97          0096  1 !
;    98          0097  1 !       020     RER00020        Ron Randall     17-Mar-1983
```

```
;   99          0098  1 !                   For DSRPLUS:  Added termination for topnotes.
;  100          0099  1 !
;  101          0100  1 !       019         REM00019        Ray Marshall    07-Mar-1983
;  102          0101  1 !                   Global edit of all modules. Updated module names, idents,
;  103          0102  1 !                   copyright dates. Changed require files to BLISS library.
;  104          0103  1 !--
;  105          0104  1
```

```
;  107          0105  1 %SBTTL 'Module Level Declarations'
;  108          0106  1 !
;  109          0107  1 ! INCLUDE FILES:
;  110          0108  1 !
;  111          0109  1 LIBRARY 'NXPORT:XPORT';                 ! XPORT Library
;  112          0110  1 REQUIRE 'REQ:RNODEF';                   ! RUNOFF variant definitions
;  113          0241  1
;  114        U 0242  1 %IF DSRPLUS %THEN
;  115        U 0243  1 LIBRARY 'REQ:DPLLIB';                 ! DSRPLUS BLISS Library
;  116          0244  1 %ELSE
;  117          0245  1 LIBRARY 'REQ:DSRLIB';                 ! DSR BLISS Library
;  118          0246  1 %FI
;  119          0247  1
;  120          0248  1 !
;  121          0249  1 ! MACROS:
;  122          0250  1 !
;  123          0251  1 MACRO
;  124          0252  1     !
;  125          0253  1     ! Close a file on a given IOB, if it was open.
;  126          0254  1     !
;  127        M 0255  1     close_if_open (which_iob) =
;  128        M 0256  1         IF  .which_iob [IOB$V_OPEN]
;  129        M 0257  1         THEN
;  130        M 0258  1             status = $XPO_CLOSE (IOB = which_iob)
;  131          0259  1     %;
;  132          0260  1
;  133          0261  1 MACRO
;  134          0262  1     !
;  135          0263  1     ! This macro makes certain that when blank lines
;  136          0264  1     ! are skipped, change bars don't get output.
;  137          0265  1     !
;  138        M 0266  1     no_bar_skip (n) =
;  139        M 0267  1         BEGIN
;  140        M 0268  1         LOCAL
;  141        M 0269  1             hold_bars;
;  142        M 0270  1
;  143        M 0271  1         hold_bars = .tsf_bars;
;  144        M 0272  1         tsf_bars  = false;
;  145        M 0273  1         uskipl (n);
;  146        M 0274  1         tsf_bars  = .hold_bars;
;  147          0275  1         END %;
;  148          0276  1 !
;  149          0277  1 ! EXTERNAL REFERENCES:
;  150          0278  1 !
;  151          0279  1 EXTERNAL
;  152          0280  1     fnct        : fnct_definition,
;  153          0281  1     fra         : FIXED_STRING,
;  154          0282  1     gca         : gca_definition,
;  155          0283  1     hct         : hct_definition,
;  156        U 0284  1 %IF DSRPLUS %THEN
;  157        U 0285  1     kwtrpn,                       ! KeyWord Table for .Reference_Point Names
;  158          0286  1 %FI
;  159          0287  1     phan        : phan_definition,
;  160          0288  1     tsf         : tsf_definition,
;  161          0289  1     brnoob      : $XPO_IOB (),
;  162          0290  1     tteiob      : $XPO_IOB (),
;  163          0291  1     ttiiob      : $XPO_IOB (),
```

```
;  164          0292  1     ttoiob      : $XPO_IOB (),
;  165          0293  1     tsiiob      : $XPO_IOB ();
;  166          0294  1
;  167        U 0295  1 %IF DSRPLUS %THEN
;  168        U 0296  1 EXTERNAL
;  169        U 0297  1     topnot      : tn_definition,
;  170        U 0298  1     tniiob      : REF $XPO_IOB (),
;  171        U 0299  1     tnuiob      : REF $XPO_IOB ();
;  172        U 0300  1
;  173        U 0301  1 EXTERNAL LITERAL
;  174        U 0302  1     rnfpc4,
;  175        U 0303  1     rnfpc5,
;  176        U 0304  1     rnfcrc,
;  177        U 0305  1     rnfcro,
;  178        U 0306  1     rnfcru,
;  179        U 0307  1     rnfcr1,
;  180        U 0308  1     rnfcr2,
;  181        U 0309  1     rnfetd;
;  182        U 0310  1
;  183        U 0311  1 EXTERNAL ROUTINE
;  184        U 0312  1     tnfil,
;  185        U 0313  1     erms,
;  186        U 0314  1     tpnote;
;  187          0315  1 %FI
;  188          0316  1
;  189          0317  1 EXTERNAL LITERAL
;  190          0318  1     rnfefd,
;  191          0319  1     rnferd,
;  192          0320  1     rnfile,
;  193          0321  1     rnfipf,
;  194          0322  1     rnfned,
;  195          0323  1     rnfpct,
;  196          0324  1     rnfpc2,
;  197          0325  1     rnfpc3;
;  198          0326  1
;  199          0327  1 EXTERNAL ROUTINE
;  200          0328  1     botpag,     clh,          cpageb,
;  201          0329  1     erm,        erml,         fn,          foobot,      fooout,
;  202        U 0330  1 %IF DSRPLUS %THEN
;  203        U 0331  1     outcref,
;  204          0332  1 %FI
;  205          0333  1     outnj,            putmsg,       remneg,      remmrg,
;  206          0334  1     toppag,     tpr,          tsttfe,      uform,       uskipl;
;  207          0335  1
;  208        U 0336  1 %IF (%BLISS (BLISS36) AND DSRPLUS) %THEN
;  209        U 0337  1 EXTERNAL ROUTINE
;  210        U 0338  1     dispsi : NOVALUE,   ttyres : NOVALUE,
;  211        U 0339  1     freref : NOVALUE;   ! Free's C-REF data structure buffers.
;  212          0340  1 %FI
;  213          0341  1
```

```
;  215     0342  1 %SBTTL 'RTERM - RUNOFF termination'
;  216     0343  1 GLOBAL ROUTINE rterm : NOVALUE =
;  217     0344  1
;  218     0345  1 !++
;  219     0346  1 ! FUNCTIONAL DESCRIPTION:
;  220     0347  1 !
;  221     0348  1 !       RUNOFF termination.
;  222     0349  1 !
;  223     0350  1 ! FORMAL PARAMETERS:    None
;  224     0351  1 !
;  225     0352  1 ! IMPLICIT INPUTS:      None
;  226     0353  1 !
;  227     0354  1 ! IMPLICIT OUTPUTS:     None
;  228     0355  1 !
;  229     0356  1 ! ROUTINE VALUE:
;  230     0357  1 ! COMPLETION CODES:     None
;  231     0358  1 !
;  232     0359  1 ! SIDE EFFECTS:         None
;  233     0360  1 !--
;  234     0361  1
;  235     0362  2     BEGIN
;  236     0363  2     LOCAL
;  237     0364  2         hold_phan_top_page,
;  238     0365  2         status;
;  239     0366  2
;  240     0367  2     tsttfe (0);                     ! Check for too few ends.
;  241     0368  2     remneg ();                      ! Check for unreported bad backdents.
;  242     0369  2     remmrg ();                      ! Check for unreported bad margin settings.
;  243     0370  2
;  244     0371  2     IF  .fnct_collecting            ! Missing .END FOOTNOTE.
;  245     0372  2     THEN
;  246     0373  3         BEGIN
;  247     0374  3         erml (rnfefd);              ! Report the error.
;  248     0375  3         fn   (h_end_footnote);      ! Terminate footnote activity.
;  249     0376  2         END;
;  250     0377  2
;  251   U 0378  2 %IF DSRPLUS %THEN
;  252   U 0379  2     !
;  253   U 0380  2     ! Missing .END TOPNOTE.
;  254   U 0381  2     !
;  255   U 0382  2     IF  .tn_collecting
;  256   U 0383  2     THEN
;  257   U 0384  2         BEGIN
;  258   U 0385  2         erml   (rnfetd);            ! Report the error.
;  259   U 0386  2         tpnote (h_end_topnote);     ! Terminate topnote activity.
;  260   U 0387  2
;  261   U 0388  2         !
;  262   U 0389  2         ! If topnote file happens to be open, close it.
;  263   U 0390  2         !
;  264   U 0391  2         IF  .tnuiob [IOB$V_OPEN]
;  265   U 0392  2         THEN
;  266   U 0393  2             BEGIN
;  267   U 0394  2             tn_type = 0;
;  268   U 0395  2             tnfil (tn_close);
;  269   U 0396  2             END;
;  270   U 0397  2
;  271   U 0398  2         IF  .tniiob [IOB$V_OPEN]
```

```
;  272     U 0399  2          THEN
;  273     U 0400  2              BEGIN
;  274     U 0401  2              tn_type = 1;
;  275     U 0402  2              tnfil (tn_close);
;  276     U 0403  2              END;
;  277     U 0404  2
;  278     U 0405  2          END;
;  279     U 0406  2
;  280     U 0407  2      !
;  281     U 0408  2      ! Delete both topnote files.
;  282     U 0409  2      !
;  283     U 0410  2      tn_type = 0;
;  284     U 0411  2      tnfil (tn_delete);
;  285     U 0412  2      tn_type = 1;
;  286     U 0413  2      tnfil (tn_delete);
;  287       0414  2  %FI
;  288       0415  2
;  289       0416  2      outnj ();
;  290       0417  2      hold_phan_top_page = .phan_top_page;        ! Remember if the user said
;  291       0418  2                                                 ! .PAGE as the last thing in
;  292       0419  2                                                 ! his .RNO file.
;  293     U 0420  2  %IF DSRPLUS %THEN
;  294     U 0421  2          ! If any cross-reference information has not been written
;  295     U 0422  2          ! to the .BRN file yet, write it now.
;  296     U 0423  2          !
;  297     U 0424  2          WHILE .tsf_cref_data NEQ 0 DO
;  298     U 0425  2              BEGIN
;  299     U 0426  2              LOCAL current_block : REF VECTOR [0];
;  300     U 0427  2              current_block = .tsf_cref_data;
;  301     U 0428  2              erms (rnfile,
;  302     U 0429  2                  CH$PTR(current_block[5]), .current_block[4]);
;  303     U 0430  2              outcref ()
;  304     U 0431  2              END;
;  305       0432  2  %FI
;  306       0433  2
;  307       0434  2      !
;  308       0435  2      ! Force out waiting footnotes, if any.
;  309       0436  2      !
;  310       0437  2      IF  .fnct_count NEQ 0
;  311       0438  2      THEN                                       ! There are footnotes to be output.
;  312       0439  3          IF  NOT .phan_paging OR (.phan_llines GEQ (2 * .phan_slines))
;  313       0440  3          THEN
;  314       0441  3              !
;  315       0442  3              ! If not paging, just put out footnotes immediately.
;  316       0443  3              ! Ditto, if the page is rather long.
;  317       0444  3              !
;  318       0445  3              BEGIN
;  319       0446  3
;  320       0447  3              !
;  321       0448  3              ! Special case:  We're at the very top of the first page.
;  322       0449  3              !
;  323       0450  3              IF  .phan_top_first
;  324       0451  3              THEN
;  325       0452  3                  toppag ();
;  326       0453  3
;  327       0454  3              fooout (.fnct_count)           ! Dump them all at once.
;  328       0455  3              END
```

```
;  329     0456  2          ELSE
;  330     0457  2               !
;  331     0458  2               ! There's a whole pile of footnotes left over, and they
;  332     0459  2               ! have to be output at the bottom of the page.  This may involve
;  333     0460  2               ! spreading them out across several pages.
;  334     0461  2               !
;  335     0462  3               BEGIN
;  336     0463  3
;  337     0464  3               !
;  338     0465  3               ! Special case:  We're at the very top of the first page.
;  339     0466  3               !
;  340     0467  3               IF  .phan_top_first
;  341     0468  3               THEN
;  342     0469  3                   toppag ();
;  343     0470  3
;  344     0471  3               !
;  345     0472  3               ! Loop over all remaining footnotes
;  346     0473  3               !
;  347     0474  3               WHILE 1 DO
;  348     0475  4                   BEGIN
;  349     0476  4                   !
;  350     0477  4                   ! First get rid of as many footnotes as possible.
;  351     0478  4                   !
;  352     0479  4                   foobot ();
;  353     0480  4
;  354     0481  4                   !
;  355     0482  4                   ! Now see if there's any more to be output.  If not, we're at
;  356     0483  4                   ! the real end of the document (almost).
;  357     0484  4                   !
;  358     0485  4                   IF  .fnct_count GTR 0
;  359     0486  4                   THEN
;  360     0487  4                       !
;  361     0488  4                       ! Because not all footnotes were output, we have to
;  362     0489  4                       ! start a new page, so that successive iterations
;  363     0490  4                       ! through this loop force out additional ones.
;  364     0491  4                       !
;  365     0492  5                       BEGIN
;  366     0493  5                       botpag ();          ! First, finish the bottom of the page
;  367     0494  5                       toppag ();          ! And now start the next page.
;  368     0495  5                       END
;  369     0496  4                   ELSE
;  370     0497  4                       !
;  371     0498  4                       ! All footnotes have been output.  Termination of this
;  372     0499  4                       ! page is handled in a special way.
;  373     0500  4                       !
;  374     0501  4                       EXITLOOP
;  375     0502  4
;  376     0503  4                   END
;  377     0504  4
;  378     0505  2               END;
;  379     0506  2
;  380     0507  2          !
;  381     0508  2          ! Force out FIGURE DEFERRED space, if any.
;  382     0509  2          !
;  383     0510  2          IF  .phan_figure NEQ 0
;  384     0511  2          THEN
;  385     0512  3              BEGIN
```

```
;  386     0513  3          !
;  387     0514  3          ! Special case:  We're at the very top of the first page.
;  388     0515  3          !
;  389     0516  3          IF  .phan_top_first
;  390     0517  3          THEN
;  391     0518  3              !
;  392     0519  3              ! Automatically dumps out the space after starting a page.
;  393     0520  3              !
;  394     0521  3              toppag ()
;  395     0522  3          ELSE
;  396     0523  3              !
;  397     0524  3              ! Dump out the space.
;  398     0525  3              !
;  399     0526  3              no_bar_skip (.phan_figure);
;  400     0527  3
;  401     0528  3          END;
;  402     0529  2
;  403     0530  2      !
;  404     0531  2      ! If the page number is to be centered at the bottom of the
;  405     0532  2      ! page then position there and output it.
;  406     0533  2      !
;  407     0534  2      IF  .hct_layoutn NEQ 0
;  408     0535  2      THEN
;  409     0536  2          cpageb ();
;  410     0537  2
;  411     0538  2      !
;  412     0539  2      ! If /PAUSE or /SIMULATE force next page before
;  413     0540  2      ! generating summary information.
;  414     0541  2      !
;  415     0542  2      IF  .phan_pause OR .phan_simulate
;  416     0543  2      THEN
;  417     0544  2          BEGIN
;  418     0545  2
;  419     0546  2          !
;  420     0547  2          ! Always force blank lines at end of document when in /SIMULATE mode.
;  421     0548  2          ! If skipping output, phan_bottom has already been set by the last
;  422     0549  2          ! page actually output.  If this is the "normal case", i.e., user didn't
;  423     0550  2          ! specify /PAGES, then this is the time to update phan_bottom, since
;  424     0551  2          ! NEWPAG won't get called to do it.
;  425     0552  2          !
;  426     0553  2          IF  NOT .gca_skip_out
;  427     0554  2          THEN
;  428     0555  2              phan_bottom = .phan_lines_tp;       ! UFORM expects phan_bottom set.
;  429     0556  2
;  430     0557  2          !
;  431     0558  2          ! Force on generation of output so blank lines get output
;  432     0559  2          !
;  433     0560  2          gca_skip_out = false;
;  434     0561  2          uform ();
;  435     0562  2          END
;  436     0563  2      ELSE
;  437     0564  2
;  438     0565  2          !
;  439     0566  2          ! Not /SIMULATE or /PAUSE
;  440     0567  2          !
;  441     0568  2          IF  .hold_phan_top_page AND (NOT .gca_cmd_quick)
;  442     0569  3
```

```
;  443          0570  3          !
;  444          0571  3          ! If user said /NOOUTPUT, don't call CLH.
;  445          0572  3          !
;  446          0573  2          THEN
;  447          0574  2              !
;  448          0575  2              ! User said .PAGE after all text was finished, intending
;  449          0576  2              ! to generate a form-feed
;  450          0577  2              !
;  451          0578  3              BEGIN
;  452          0579  3              fs_init (fra);
;  453          0580  3              fs_wchar (fra, %O'14');      ! Insert a form feed
;  454          0581  3              clh (clh_out_nocrlf)
;  455          0582  2              END;
;  456          0583  2
;  457        U 0584  2 %IF DSRPLUS %THEN
;  458        U 0585  2     IF  .gca_cref_changed          ! Did one change on last run?
;  459        U 0586  2     THEN
;  460        U 0587  2         BEGIN
;  461        U 0588  2
;  462        U 0589  2 %IF %BLISS (BLISS32) %THEN
;  463        U 0590  2 !+
;  464        U 0591  2 ! This logic increments the rerun count if necessary. It assures that the
;  465        U 0592  2 ! file will be rerun only if the user said /AUTO. On Pass 1, if no previous
;  466        U 0593  2 ! .BRN file existed, the rerun count has already been incremented (by DOOPTS),
;  467        U 0594  2 ! so we do not increment again.
;  468        U 0595  2 !-
;  469        U 0596  2         IF  .gca_black_box
;  470        U 0597  2            AND
;  471        U 0598  2             (  (.gca_pass_count NEQ 1)
;  472        U 0599  2                          OR
;  473        U 0600  2                .gca_old_brn_exists )
;  474        U 0601  2         THEN
;  475        U 0602  2             gca_rerun_count = .gca_rerun_count + 1;
;  476        U 0603  2
;  477        U 0604  2         IF .gca_pass_count EQL 4
;  478        U 0605  2         THEN
;  479        U 0606  2             BEGIN
;  480        U 0607  2             putmsg (rnfcro, 0, 0);       ! Tell user cross-refs are oscillating,
;  481        U 0608  2             putmsg (rnfcr1, 0, 0);       !  ... and how he can find out where.
;  482        U 0609  2             END
;  483        U 0610  2         ELSE
;  484        U 0611  2
;  485        U 0612  2             IF .gca_pass_count GTR .gca_rerun_count
;  486        U 0613  2             THEN
;  487        U 0614  2                 BEGIN
;  488        U 0615  2 %FI
;  489        U 0616  2                 putmsg (rnfcrc, 0, 0);   ! Tell user forward cross-refs changed,
;  490        U 0617  2                 putmsg (rnfcr1, 0, 0);   !  ... and what he can do
;  491        U 0618  2                 putmsg (rnfcr2, 0, 0);   !  ... about it.
;  492        U 0619  2
;  493        U 0620  2 %IF %BLISS (BLISS32) %THEN
;  494        U 0621  2                 END;
;  495        U 0622  2 %FI
;  496        U 0623  2
;  497        U 0624  2         END;
;  498        U 0625  2
;  499        U 0626  2     IF .gca_cref_errcnt  GTR 0
```

```
;  500        U 0627  2     THEN
;  501        U 0628  2         BEGIN
;  502        U 0629  2         putmsg (rnfcru, .gca_cref_errcnt,  0); ! Unresolved refs
;  503        U 0630  2         IF NOT .gca_black_box
;  504        U 0631  2         THEN
;  505        U 0632  2             putmsg (rnfcr1, 0, 0);              !  ... and what he can do
;  506        U 0633  2         END;
;  507          0634  2 %FI
;  508          0635  2     !
;  509          0636  2     ! If the user specified /PAGES but no pages were output then tell him so.
;  510          0637  2     !
;  511          0638  2     IF .gca_cmd_pages
;  512          0639  2         AND
;  513          0640  3     (.phan_lines_tp EQL 0)
;  514          0641  2         AND
;  515          0642  3     (.gca_page_cnt EQL 0)
;  516          0643  2     THEN
;  517          0644  2         putmsg (rnfipf, 0, 0);
;  518          0645  2
;  519          0646  2     IF  .gca_fehler
;  520          0647  2     THEN                                    ! If any errors were detected,
;  521          0648  2         putmsg (rnferd, .gca_errcnt, 0)     ! report # of lines of error messages.
;  522          0649  2     ELSE                                    ! If no errors detected and
;  523          0650  2
;  524          0651  2         IF  .termination_log
;  525          0652  2         THEN                                ! /LOG was specified,
;  526          0653  2             putmsg (rnfned, 0, 0);          ! then just identify version and
;  527          0654  2                                             ! report 0 diagnostics.
;  528          0655  2
;  529          0656  2     !
;  530          0657  2     ! The following IF supplies the following logic:  If /NOOUT was specified,
;  531          0658  2     ! this block isn't done.  If not, and either /LOG was specified or errors
;  532          0659  2     ! were encountered, then perform the block.
;  533          0660  2     !
;  534          0661  3     IF  (.termination_log OR .gca_fehler)
;  535          0662  2     THEN                                    ! If so, then report pages.
;  536          0663  2         !
;  537          0664  2         ! This block determines whether or not there is currently a valid page
;  538          0665  2         ! count. If not, it attempts to calculate one based on the number of
;  539          0666  2         ! lines output.
;  540          0667  2         !
;  541          0668  2         ! Then it writes the second line of the termination message to the
;  542          0669  2         ! log device. This line reports the number of pages output and
;  543          0670  2         ! idintifies the output filename.
;  544          0671  2         !
;  545          0672  3         BEGIN
;  546          0673  3
;  547          0674  3         !
;  548          0675  3         ! If this is a document broken up into pages, then count logical pages.
;  549          0676  3         ! Otherwise, (i.e. .NO PAGING) figure out # of spooled pages this is.
;  550          0677  3         !
;  551          0678  3         IF NOT .gca_cmd_quick
;  552          0679  3         THEN
;  553          0680  4             BEGIN
;  554          0681  5             IF  (NOT .phan_paging)
;  555          0682  4             THEN
;  556          0683  4                 !
```

```
;  557     0684  4                    ! Not a document organized into pages.
;  558     0685  4                    ! Assume output device spooler will break it up.
;  559     0686  4                    !
;  560     0687  4                    gca_page_cnt = .gca_page_cnt +
;  561     0688  4                                   (.phan_lines_tp + .phan_slines - 1) / .phan_slines
;  562     0689  4                ELSE
;  563     0690  4
;  564     0691  4                    !
;  565     0692  4                    ! Not .NO PAGING, so bump count only if last page is not empty.
;  566     0693  4                    !
;  567     0694  5                    IF  (.phan_lines_tp GTR 0)
;  568     0695  4                        AND
;  569     0696  5                        (NOT .gca_skip_out  OR .gca_page_cnt GTR 0)
;  570     0697  4                    THEN
;  571     0698  4                        gca_page_cnt = .gca_page_cnt + 1;
;  572     0699  4
;  573     0700  4                !
;  574     0701  4                ! Report how many pages generated and to what file.
;  575     0702  4                !
;  576     0703  4                putmsg (rnfpct, .gca_page_cnt, 0);
;  577     0704  3                END;
;  578     0705  3
;  579     0706  3            !
;  580     0707  3            ! Report on what information, if any, was written to a .BRN file.
;  581     0708  3            ! (This is reported whether or not Quick processing was in effect.)
;  582     0709  3            ! If doing cross-referencing, also report how many cross-referencing
;  583     0710  3            ! hits were read from the old (input) .BRN file.
;  584     0711  3            !
;  585     0712  3            SELECT true OF
;  586     0713  3                SET
;  587     0714  3                [.gca_index_cnt    GTR 0] : putmsg (rnfpc2, .gca_index_cnt,    0);
;  588     0715  3                [.gca_contents_cnt GTR 0] : putmsg (rnfpc3, .gca_contents_cnt, 0);
;  589     0716  3
;  590   U 0717  3 %IF DSRPLUS %THEN
;  591   U 0718  3                [.gca_new_cref_cnt GTR 0] : putmsg (rnfpc4, .gca_new_cref_cnt, 0);
;  592   U 0719  3                [.gca_old_cref_cnt GTR 0] : putmsg (rnfpc5, .gca_old_cref_cnt, 0);
;  593     0720  3 %FI
;  594     0721  3
;  595     0722  3                TES;
;  596     0723  3
;  597     0724  2            END;
;  598     0725  2
;  599   U 0726  2 %IF (%BLISS (BLISS36) AND DSRPLUS) %THEN
;  600   U 0727  2            !+
;  601   U 0728  2            ! Call TTYRES to restore terminal to its original mode (don't pass
;  602   U 0729  2            ! escape sequences, and reset wrapping width). This call is made only
;  603   U 0730  2            ! if the user said /DEC=VT100, which caused these terminal characteristics
;  604   U 0731  2            ! to be modified. In this case, the saved terminal width in GCA_TWIDTH_36
;  605   U 0732  2            ! will be different than its initialized value of -1.
;  606   U 0733  2            !-
;  607   U 0734  2            dispsi ();                          ! Disable the PSI CTRL/C interrupt.
;  608   U 0735  2
;  609   U 0736  2            IF  .gca_twidth_36 NEQ -1
;  610   U 0737  2            THEN
;  611   U 0738  2                ttyres ();
;  612     0739  2 %FI
;  613     0740  2
```

```
;  614          0741  2      !
;  615          0742  2      ! Close all files.
;  616          0743  2      !
;  617          0744  2      close_if_open (brnoob);                 ! Close intermediate binary file.
;  618          0745  2      close_if_open (tteiob);
;  619          0746  2      close_if_open (ttiiob);
;  620          0747  2      close_if_open (ttoiob);
;  621          0748  2      close_if_open (tsiiob);
;  622          0749  2
;  623        U 0750  2  %IF (%BLISS (BLISS36) AND DSRPLUS) %THEN
;  624        U 0751  2      !+
;  625        U 0752  2      ! If there any entries in the cross-ref structure, make sure we release
;  626        U 0753  2      ! all of that dynamic memory and clear out the listhead's pointer.  This
;  627        U 0754  2      ! is necessary because of the interactive way DSR/PLUS is used on TOPS-20.
;  628        U 0755  2      ! i.e. when RUNOFF is reentered, we don't want the previous run's text
;  629        U 0756  2      ! lying around.
;  630        U 0757  2      !-
;  631        U 0758  2          BEGIN
;  632        U 0759  2          LOCAL
;  633        U 0760  2              current_entry : REF VECTOR[0],
;  634        U 0761  2              desc : REF $STR_DESCRIPTOR( CLASS=DYNAMIC ), ! string desc.
;  635        U 0762  2              temp_pointer;
;  636        U 0763  2
;  637        U 0764  2          current_entry = .kwtrpn;
;  638        U 0765  2
;  639        U 0766  2          WHILE current_entry[0] NEQ 0 DO
;  640        U 0767  2              BEGIN
;  641        U 0768  2              freref (.current_entry[2]);             ! Release old c-ref bfrs,
;  642        U 0769  2              desc = .current_entry[1];
;  643        U 0770  2              $XPO_FREE_MEM (  STRING = (  .desc [str$h_length]  ! ref. name,
;  644        U 0771  2                                          ,.desc [str$a_pointer] ) );
;  645        U 0772  2              $XPO_FREE_MEM (  BINARY_DATA =( 2 ! descriptor for name,
;  646        U 0773  2                                          ,.current_entry[1] ) );
;  647        U 0774  2              temp_pointer = .current_entry[0];
;  648        U 0775  2              $XPO_FREE_MEM (  BINARY_DATA =( 3 ! and keyword block
;  649        U 0776  2                                          ,current_entry[0] ) );
;  650        U 0777  2              current_entry = .temp_pointer
;  651        U 0778  2              END;
;  652        U 0779  2          kwtrpn = 0              ! initialize listhead
;  653        U 0780  2          END;
;  654          0781  2  %FI
;  655          0782  2      RETURN;
;  656          0783  1      END;                                        ! End of RTERM.


                                .TITLE  RTERM RUNOFF termination and cleanup
                                .IDENT  \V04-000\

                                .EXTRN  FNCT, FRA, GCA, HCT
                                .EXTRN  PHAN, TSF, BRNOOB
                                .EXTRN  TTEIOB, TTIIOB, TTOIOB
                                .EXTRN  TSIIOB, RNFEFD, RNFERD
                                .EXTRN  RNFILE, RNFLSF, RNFNED
                                .EXTRN  RNFPCT, RNFPC2, RNFPC3
                                .EXTRN  BOTPAG, CLH, CPAGEB
                                .EXTRN  ERM, ERML, FN, FOOBOT
                                .EXTRN  FOOOUT, OUTNJ, PUTMSG
```

```
                                                .EXTRN  REMNEG, REMMRG, TOPPAG
                                                .EXTRN  TPR, TSTTFE, UFORM
                                                .EXTRN  USKIPL, XPO$CLOSE
                                                .EXTRN  XPO$FAILURE

                                                .PSECT  $CODE$,NOWRT,2

                                    0FFC 00000  .ENTRY  RTERM, Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11 ; 0343
                              5B 00000000G  EF   9E 00002  MOVAB  FNCT, R11                          ;
                              5A 00000000G  EF   9E 00009  MOVAB  XPO$CLOSE, R10                     ;
                              59 00000000G  EF   9E 00010  MOVAB  XPO$FAILURE, R9                    ;
                              58 00000000G  EF   9E 00017  MOVAB  PUTMSG, R8                         ;
                              57 00000000G  EF   9E 0001E  MOVAB  FRA+4, R7                          ;
                              56 00000000G  EF   9E 00025  MOVAB  GCA+208, R6                        ;
                              55 00000000G  EF   9E 0002C  MOVAB  PHAN+12, R5                        ;
                                            7E   D4 00033  CLRL   -(SP)                              ; 0367
                   00000000G  EF              01   FB 00035  CALLS  #1, TSTTFE                      ;
                   00000000G  EF              00   FB 0003C  CALLS  #0, REMNEG                      ; 0368
                   00000000G  EF              00   FB 00043  CALLS  #0, REMMRG                      ; 0369
                              16        14    AB   E9 0004A  BLBC   FNCT+20, 1$                     ; 0371
                                 00000000G  8F   DD 0004E  PUSHL  #RNFEFD                          ; 0374
                   00000000G  EF              01   FB 00054  CALLS  #1, ERML                        ;
                                            39   DD 0005B  PUSHL  #57                               ; 0375
                   00000000G  EF              01   FB 0005D  CALLS  #1, FN                          ;
                   00000000G  EF              00   FB 00064  1$:   CALLS  #0, OUTNJ                 ; 0416
                              54        F4    A5   D0 0006B  MOVL   PHAN, HOLD_PHAN_TOP_PAGE         ; 0417
                                            6B   D5 0006F  TSTL   FNCT                              ; 0437
                                            44   13 00071  BEQL   7$
                              0B        1C    B5   E9 00073  BLBC   @PHAN+40, 2$                    ; 0439
                        50        18    A5   01   78 00077  ASHL   #1, PHAN+36, R0                  ;
                              50        F8    B5   D1 0007C  CMPL   @PHAN+4, R0                     ;
                                            16   19 00080  BLSS   4$
                              07        0C    A5   E9 00082  2$:   BLBC   PHAN+24, 3$               ; 0450
                   00000000G  EF              00   FB 00086  CALLS  #0, TOPPAG                      ; 0452
                                            6B   DD 0008D  3$:   PUSHL  FNCT                        ; 0454
                   00000000G  EF              01   FB 0008F  CALLS  #1, FOOOUT                     ;
                                            1F   11 00096  BRB    7$
                              07        0C    A5   E9 00098  4$:   BLBC   PHAN+24, 6$               ; 0467
                   00000000G  EF              00   FB 0009C  5$:   CALLS  #0, TOPPAG                ; 0469
                   00000000G  EF              00   FB 000A3  6$:   CALLS  #0, FOOBOT                ; 0479
                                            6B   D5 000AA  TSTL   FNCT                              ; 0485
                                            09   15 000AC  BLEQ   7$
                   00000000G  EF              00   FB 000AE  CALLS  #0, BOTPAG                      ; 0493
                                            E5   11 000B5  BRB    5$                                ; 0494
                              52        04    A5   D0 000B7  7$:   MOVL   PHAN+16, R2                ; 0510
                                            34   13 000BB  BEQL   9$
                              09        0C    A5   E9 000BD  BLBC   PHAN+24, 8$                     ; 0517
                   00000000G  EF              00   FB 000C1  CALLS  #0, TOPPAG                      ; 0522
                                            27   11 000C8  BRB    9$
                              50 00000000G  EF   D0 000CA  8$:   MOVL   TSF, R0                      ; 0527
                        53        7C    A0   01   00   EF 000D1  EXTZV  #0, #1, 124(R0), HOLD_BARS   ;
                                    7C    A0   01   8A 000D7  BICB2  #1, 124(R0)                     ;
                                            52   DD 000DB  PUSHL  R2                                ;
                   00000000G  EF              01   FB 000DD  CALLS  #1, USKIPL                     ;
                              50 00000000G  EF   D0 000E4  MOVL   TSF, R0                           ;
                        7C    A0        01    00   53   F0 000EB  INSV   HOLD_BARS, #0, #1, 124(R0) ;
                                 00000000G  EF   D5 000F1  9$:   TSTL   HCT+32                       ; 0535
```

```
                                    07  13 000F7           BEQL   10$                          ;
        00000000G  EF               00  FB 000F9           CALLS  #0, CPAGEB                   ; 0537
                   04          30   A5  E8 00100  10$:     BLBS   PHAN+60, 11$                 ; 0543
                   14          28   A5  E9 00104           BLBC   PHAN+52, 13$                 ;
                   04          A0   A6  E8 00108  11$:     BLBS   GCA+112, 12$                 ; 0554
               2C  A5               65  D0 0010C           MOVL   PHAN+12, PHAN+56             ; 0556
                               A0   A6  D4 00110  12$:     CLRL   GCA+112                      ; 0561
        00000000G  EF               00  FB 00113           CALLS  #0, UFORM                    ; 0562
                                    24  11 0011A           BRB    14$                          ; 0543
                   21               54  E9 0011C  13$:     BLBC   HOLD_PHAN_TOP_PAGE, 14$      ; 0569
                   1E               66  E8 0011F           BLBS   GCA+208, 14$                 ;
                               08   A7  D4 00122           CLRL   FRA+12                       ; 0579
               FC  A7          0C   A7  9E 00125           MOVAB  FRA+16, FRA                  ;
                   67          FC   A7  D0 0012A           MOVL   FRA, FRA+4                   ;
               00  B7               0C  90 0012E           MOVB   #12, @FRA+4                  ; 0580
                                    67  D6 00132           INCL   FRA+4                        ;
                               08   A7  D6 00134           INCL   FRA+12                       ;
                                    0B  DD 00137           PUSHL  #11                          ; 0581
        00000000G  EF               01  FB 00139           CALLS  #1, CLH                      ;
     14        01  A6               02  E1 00140  14$:     BBC    #2, GCA+209, 15$             ; 0638
                                    65  D5 00145           TSTL   PHAN+12                      ; 0640
                                    10  12 00147           BNEQ   15$                          ;
                               88   A6  D5 00149           TSTL   GCA+88                       ; 0642
                                    0B  12 0014C           BNEQ   15$                          ;
                                    7E  7C 0014E           CLRQ   -(SP)                        ; 0644
                        00000000G   8F  DD 00150           PUSHL  #RNFIPF                      ;
                   68               03  FB 00156           CALLS  #3, PUTMSG                   ;
                   0D          F0   A6  E9 00159  15$:     BLBC   GCA+192, 16$                 ; 0646
                                    7E  D4 0015D           CLRL   -(SP)                        ; 0648
                               F4   A6  DD 0015F           PUSHL  GCA+196                      ;
                        00000000G   8F  DD 00162           PUSHL  #RNFERD                      ;
                                    0C  11 00168           BRB    17$                          ;
     0F            66               01  E1 0016A  16$:     BBC    #1, GCA+208, 18$             ; 0651
                                    7E  7C 0016E           CLRQ   -(SP)                        ; 0653
                        00000000G   8F  DD 00170           PUSHL  #RNFNED                      ;
                   68               03  FB 00176  17$:     CALLS  #3, PUTMSG                   ;
     04            66               01  E0 00179           BBS    #1, GCA+208, 19$             ; 0661
                   5C          F0   A6  E9 0017D  18$:     BLBC   GCA+192, 25$                 ;
                   33               66  E8 00181  19$:     BLBS   GCA+208, 23$                 ; 0678
                   11          1C   B5  E8 00184           BLBS   @PHAN+40, 20$                ; 0681
     50            65          18   A5  C1 00188           ADDL3  PHAN+36, PHAN+12, R0         ; 0688
                                    50  D7 0018D           DECL   R0                           ;
                   50          18   A5  C6 0018F           DIVL2  PHAN+36, R0                  ;
               88  A6               50  C0 00193           ADDL2  R0, GCA+88                   ;
                                    10  11 00197           BRB    22$                          ; 0687
                                    65  D5 00199  20$:     TSTL   PHAN+12                      ; 0694
                                    0C  15 0019B           BLEQ   22$                          ;
                   05          A0   A6  E9 0019D           BLBC   GCA+112, 21$                 ; 0696
                               88   A6  D5 001A1           TSTL   GCA+88                       ;
                                    03  15 001A4           BLEQ   22$                          ;
                               88   A6  D6 001A6  21$:     INCL   GCA+88                       ; 0698
                                    7E  D4 001A9  22$:     CLRL   -(SP)                        ; 0703
                               88   A6  DD 001AB           PUSHL  GCA+88                       ;
                        00000000G   8F  DD 001AE           PUSHL  #RNFPCT                      ;
                   68               03  FB 001B4           CALLS  #3, PUTMSG                   ;
                   50          90   A6  D0 001B7  23$:     MOVL   GCA+96, R0                   ; 0714
                                    0D  15 001BB           BLEQ   24$                          ;
```

```
                                 7E  D4 001BD          CLRL    -(SP)                                                          ;
                                 50  DD 001BF          PUSHL   R0                                                             ;
                      00000000G  8F  DD 001C1          PUSHL   #RNFPC2                                                        ;
                             68      03 001C7          CALLS   #3, PUTMSG                                                     ;
                             50   8C A6 D0 001CA 24$:   MOVL    GCA+92, R0                                                     ; 0715
                                  0D 15 001CE          BLEQ    25$
                                 7E  D4 001D0          CLRL    -(SP)                                                          ;
                                 50  DD 001D2          PUSHL   R0                                                             ;
                      00000000G  8F  DD 001D4          PUSHL   #RNFPC3                                                        ;
                             68      03 001DA          CALLS   #3, PUTMSG                                                     ;
                             14 00000000G EF E9 001DD 25$:   BLBC    BRNOOB+50, 26$                                         ; 0744
                00000000G    EF          02 90 001E4          MOVB    #2, IOB$+44
                                 59  DD 001EB          PUSHL   R9                                                             ;
                                 7E  D4 001ED          CLRL    -(SP)                                                          ;
                      00000000G  EF  9F 001EF          PUSHAB  IOB$                                                           ;
                             6A      03 001F5          CALLS   #3, XPO$CLOSE                                                  ;
                             14 00000000G EF E9 001F8 26$:   BLBC    TTEIOB+50, 27$                                         ; 0745
                00000000G    EF          02 90 001FF          MOVB    #2, IOB$+44
                                 59  DD 00206          PUSHL   R9                                                             ;
                                 7E  D4 00208          CLRL    -(SP)                                                          ;
                      00000000G  EF  9F 0020A          PUSHAB  IOB$                                                           ;
                             6A      03 00210          CALLS   #3, XPO$CLOSE                                                  ;
                             14 00000000G EF E9 00213 27$:   BLBC    TTIIOB+50, 28$                                         ; 0746
                00000000G    EF          02 90 0021A          MOVB    #2, IOB$+44
                                 59  DD 00221          PUSHL   R9                                                             ;
                                 7E  D4 00223          CLRL    -(SP)                                                          ;
                      00000000G  EF  9F 00225          PUSHAB  IOB$                                                           ;
                             6A      03 0022B          CALLS   #3, XPO$CLOSE                                                  ;
                             14 00000000G EF E9 0022E 28$:   BLBC    TTOIOB+50, 29$                                         ; 0747
                00000000G    EF          02 90 00235          MOVB    #2, IOB$+44
                                 59  DD 0023C          PUSHL   R9                                                             ;
                                 7E  D4 0023E          CLRL    -(SP)                                                          ;
                      00000000G  EF  9F 00240          PUSHAB  IOB$                                                           ;
                             6A      03 00246          CALLS   #3, XPO$CLOSE                                                  ;
                             14 00000000G EF E9 00249 29$:   BLBC    TSIIOB+50, 30$                                         ; 0748
                00000000G    EF          02 90 00250          MOVB    #2, IOB$+44
                                 59  DD 00257          PUSHL   R9                                                             ;
                                 7E  D4 00259          CLRL    -(SP)                                                          ;
                      00000000G  EF  9F 0025B          PUSHAB  IOB$                                                           ;
                             6A      03 00261          CALLS   #3, XPO$CLOSE                                                  ;
                                     04 00264 30$:    RET                                                                    ; 0783

; Routine Size:  613 bytes,    Routine Base:  $CODE$ + 0000

;  657          0784  1
;  658          0785  1 END                                     ! End of module.
;  659          0786  0 ELUDOM



;                                PSECT SUMMARY
;
;       Name                    Bytes                           Attributes
```

```
;
;  $CODE$                          613  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        81        13        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$:RTERM/OBJ=OBJ$:RTERM MSRC$:RTERM/UPDATE=(ENH$:RTERM)

; Size:          613 code + 0 data bytes
; Run Time:          00:15.0
; Elapsed Time:      00:36.5
; Lines/CPU Min:      3148
; Lexemes/CPU-Min: 28445
; Memory Used:  189 pages
; Compilation Complete
```

RSDERM
LIS

RTERM
LIS

RNODEF
LIS

RUNOFF
LIS

RNOVMS
LIS

RSKIPS
LIS

RNODAT
LIS

SAVRES
LIS

RNOURS
LIS

RNFERM
LIS