


```

000000    UU    UU    TTTTTTTTTT    CCCCCCCC    HH    HH    AAAAAA
000000    UU    UU    TTTTTTTTTT    CCCCCCCC    HH    HH    AAAAAA
00      00    UU    UU    TT          CC          HH    HH    AA      AA
00      00    UU    UU    TT          CC          HH    HH    AA      AA
00      00    UU    UU    TT          CC          HH    HH    AA      AA
00      00    UU    UU    TT          CC          HH    HH    AA      AA
00      00    UU    UU    TT          CC          HH    HH    AA      AA
00      00    UU    UU    TT          CC          HH    HH    AA      AA
00      00    UU    UU    TT          CC          HH    HH    AA      AA
00      00    UU    UU    TT          CC          HH    HH    AA      AA
00      00    UU    UU    TT          CC          HH    HH    AA      AA
000000    UUUUUUUUUU    TT          CCCCCCCC    HH    HH    AAAAAA
000000    UUUUUUUUUU    TT          CCCCCCCC    HH    HH    AAAAAA

```

```

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
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40

```

0001 0 %TITLE 'Processor for chapter and appendix headers.'
0002 0 MODULE outcha ( IDENT = 'V04-000'
P 0003 0      %BLISS32L, ADDRESSING_MODE (EXTERNAL = LONG_RELATIVE,
0004 0      NONEXTERNAL = LONG_RELATIVE)
0005 0      ) =
0006 1 BEGIN
0007 1
0008 1 *****
0009 1 *
0010 1 *  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0011 1 *  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0012 1 *  ALL RIGHTS RESERVED.
0013 1 *
0014 1 *  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0015 1 *  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0016 1 *  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0017 1 *  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0018 1 *  OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0019 1 *  TRANSFERRED.
0020 1 *
0021 1 *  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0022 1 *  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0023 1 *  CORPORATION.
0024 1 *
0025 1 *  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0026 1 *  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0027 1 *
0028 1 *
0029 1 *****
0030 1
0031 1
0032 1 ++
0033 1 FACILITY:      DSR (Digital Standard RUNOFF) / DSRPLUS
0034 1
0035 1 ABSTRACT:      Processor for chapter and appendix headers.
0036 1
0037 1 ENVIRONMENT:   Transportable
0038 1
0039 1 AUTHOR:        K.A. ALDEN      CREATION DATE:  February, 1983
0040 1

```

```

42 0041 1 %SBTTL 'Revision History'
43 0042 1   MODIFIED BY:
44 0043 1
45 0044 1   019   KFA00019   Ken Alden   14-Sep-1983
46 0045 1   Added emphasis support for autotitle if user used STCH or
47 0046 1   STAX for emphasis. This is for DSRPLUS only.
48 0047 1
49 0048 1   018   KFA00018   Ken Alden   23-Aug-1983
50 0049 1   Removed eight lines of code that are only needed for
51 0050 1   autosubtitle. These lines were left over from OUTHDR
52 0051 1   and prevented autotitle from carrying over any emphasis
53 0052 1   the user may have given with the chapter/appendix directives.
54 0053 1
55 0054 1   017   KFA00017   Ken Alden   28-Jul-1983
56 0055 1   Tweaked the flip bit from #16.
57 0056 1
58 0057 1   016   KFA00016   Ken Alden   27-Jul-1983
59 0058 1   Fixed logic for flip that was put in during #14.
60 0059 1
61 0060 1   015   KFA00015   Ken Alden   28-Jun-1983
62 0061 1   Made all calls to this module force a "chapter-oriented" book.
63 0062 1   Unnumbered chapters will now also reset the page number to one.
64 0063 1
65 0064 1   014   KAD00014   Keith Dawson 18-May-1983
66 0065 1   Removed call to PUTTPG for FLIP output.
67 0066 1
68 0067 1   013   KFA00013   Ken Alden   4-May-1983
69 0068 1   Un-numbered chapter/appendixes now will not change the
70 0069 1   "chapter" status or page numbers of the document.
71 0070 1
72 0071 1   012   KFA00012   Ken Alden   2-May-1983
73 0072 1   Altered the way subtitles are cleared at the start of
74 0073 1   a chapter. The way it works now is: 1)AUTOTITLE is first
75 0074 1   checked. If true, then clear the subtitle. 2) If AUTOTITLE
76 0075 1   is false, the subtitle is clear anyway if AUTOSUBTITLE is true.
77 0076 1
78 0077 1   011   KFA00011   Ken Alden   29-Apr-1983
79 0078 1   Added code for the STARTODD bit;
80 0079 1
81 0080 1   010   REM00010   Ray Marshall April-1983
82 0081 1   Conditionalized for DSRPLUS so it can be used by DSR, too.
83 0082 1
84 0083 1   009   RER00009   Ron Randall  07-Apr-1983
85 0084 1   For DSRPLUS: Initialize footnote number.
86 0085 1
87 0086 1   008   KFA00008   Ken Alden   16-Mar-1983
88 0087 1   PUSH/POP_SCA now visible to DSR.
89 0088 1
90 0089 1   007   KFA00007   Ken Alden   9-Mar-1983
91 0090 1   TOC action was changed so the footnote MRA is always
92 0091 1   passed to the TOC as well as always being read. If
93 0092 1   AUTOTITLE is on, then the title MRA is also read. This
94 0093 1   passes the bolding/underlining information to the BRN, regardless.
95 0094 1
96 0095 1   006   KAD00006   Keith Dawson 07-Mar-1983
97 0096 1   Global edit of all modules. Updated module names, idents,
98 0097 1   copyright dates. Changed require files to BLISS library.

```



```

: 102      C100 1 %SBTTL 'Module Level Declarations'
: 103      0101 1
: 104      0102 1 : TABLE OF CONTENTS:
: 105      0103 1
: 106      0104 1 FORWARD ROUTINE
: 107      0105 1   OUTCHA : NOVALUE;           ! Generate the CHAPTER or APPENDIX header
: 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 : EQUATED SYMBOLS:
: 122      0250 1
: 123      0251 1 EXTERNAL LITERAL
: 124      0252 1   RINTES : UNSIGNED (8);
: 125      0253 1
: 126      0254 1 EXTERNAL LITERAL
: 127      0255 1   S_FMRA;                   !Allocated length for footnote MRA.
: 128      0256 1
: 129      0257 1 : OWN STORAGE:
: 130      0258 1
: 131      0259 1
: 132      0260 1
: 133      0261 1 : MACROS:
: 134      0262 1
: 135      0263 1 MACRO                       ! NOTE: this code depends on the arrangement
: 136      M 0264 1   show_counter =         ! of literals in TOCRTY.
: 137      M 0265 1   (
: 138      M 0266 1     (.caption_minor_type MOD 3) EQL 2
: 139      M 0267 1     OR
: 140      M 0268 1     .caption_minor_type EQL min_chapt
: 141      M 0269 1     OR
: 142      M 0270 1     .caption_minor_type EQL min_append
: 143      M 0271 1   )
: 144      0272 1   %;
: 145      0273 1
: 146      0274 1 OWN
: 147      0275 1   PP_SCA · $H_R_SCA_BLOCK;  !Used in PUSH_SCA, POP_SCA macros (defined in SCA.REQ).
: 148      0276 1
: 149      0277 1 : EXTERNAL REFERENCES:
: 150      0278 1
: 151      0279 1 EXTERNAL
: 152      0280 1   ECC : $ECC_BLOCKVECTOR,   ! Counters and display codes for Examples, Figures, Tables
: 153      0281 1   FNCT : FNCT_DEFINITION,
: 154      0282 1   FOOMRA : FIXED_STRING,
: 155      0283 1   FOOTSF : VECTOR [TSF_SIZE],
: 156      0284 1   FS01 : FIXED_STRING,
: 157      0285 1   GCA : GCA_DEFINITION,
: 158      0286 1   HCT : HCT_DEFINITION,

```

```

: 159 0287 1 HLDSP : VECTOR [MAX LEVELS],
: 160 0288 1 HLLIST : COUNTED LIST,
: 161 0289 1 IRA : FIXED STRING,
: 162 0290 1 MRA : REF FIXED STRING,
: 163 0291 1 TITMRA : FIXED STRING,
: 164 0292 1 TITTSF : VECTOR,
: 165 0293 1 NPAGEN : PAGE DEFINITION,
: 166 0294 1 PAGEN : PAGE DEFINITION,
: 167 0295 1 PHAN : PHAN DEFINITION,
: 168 0296 1 SBTMRA : FIXED STRING,
: 169 0297 1 SBTTSF : VECTOR,
: 170 0298 1 SCA : SCA_DEFINITION,
: 171 0299 1 TSF : TSF_DEFINITION;
: 172 0300 1
: 173 0301 1 EXTERNAL
: 174 0302 1 KHAR;
: 175 0303 1
: 176 0304 1 EXTERNAL ROUTINE
: 177 0305 1 endchr, endwrđ, gcpage, gcskip,
: 178 0306 1 gtpc, guskip, outcrg, outctr,
: 179 0307 1 outnj, pacsec, pacxxx, putcnt,
: 180 0308 1 puttpg, puttxt, remmrg, remneg,
: 181 0309 1 rskips, scant, sdxy, setcas,
: 182 0310 1 titles, tstblk;
: 183 0311 1

```

```

185 0312 1 %SBTTL 'OUTCHA -- Process chapter & appendix headers'
186 0313 1 GLOBAL ROUTINE outcha
187 0314 1 (
188 0315 1     lines_before,          lines_after,          test_page_amount,
189 0316 1     counter_major_type,   counter_minor_type,
190 0317 1     counter_value,       counter_display_code, counter_spaces_after,
191 0318 1     counter_pre_string_length, counter_pre_string_ptr,
192 0319 1     counter_post_string_length, counter_post_string_ptr,
193 0320 1     caption_major_type,  caption_minor_type,  caption_case,
194 0321 1     caption_is_centered, caption_is_flush_right, caption_is_run_in,
195 0322 1     caption_is_bold,    caption_is_underlined, put_into_mem_file,
196 0323 1     autosubtitle,      brn_open,            break_before_caption,
197 0324 1     lines_between,     new_page,            startodd,
198 0325 1     tocpage
199 0326 1 ) : NOVALUE =
200 0327 1
201 0328 1 +-+
202 0329 1 FUNCTIONAL DESCRIPTION:
203 0330 1
204 0331 1     Processor for chapter and appendix headers.
205 0332 1
206 0333 1 FORMAL PARAMETERS:
207 0334 1
208 0335 1 LINES_BEFORE          leave this many blank lines before the header
209 0336 1 LINES_BETWEEN        leave n blank lines between 'CHAPTER' or 'APPENDIX' & caption.
210 0337 1 LINES_AFTER          leave this many blank lines after the (non-run-in) header
211 0338 1 TEST_PAGE_AMOUNT     start a new page unless this many lines remain
212 0339 1 BREAK_BEFORE_CAPTION start a new line between the counter and caption.
213 0340 1 COUNTER_MAJOR_TYPE   (not implemented) always MAJ_RUNOFF
214 0341 1 COUNTER_MINOR_TYPE   specifies type of header: HL, Example, Figure, Table
215 0342 1 COUNTER_VALUE        (not used for Hls) numerical value of the counter
216 0343 1 COUNTER_DISPLAY_CODE (not used for Hls) display code of the counter
217 0344 1 COUNTER_SPACES_AFTER leave this many spaces between counter and caption
218 0345 1 COUNTER_PRE_STRING_LENGTH (not used for Hls) length of pre-counter string
219 0346 1 COUNTER_PRE_STRING_PTR (not used for Hls) pointer to pre-counter string
220 0347 1 COUNTER_POST_STRING_LENGTH (not used for Hls) length of post-counter string
221 0348 1 COUNTER_POST_STRING_PTR (not used for Hls) pointer to post-counter string
222 0349 1 CAPTION_MAJOR_TYPE   (not implemented) always MAJ_RUNOFF
223 0350 1 CAPTION_MINOR_TYPE   specifies whether or not to show the counter
224 0351 1 CAPTION_CASE         code for the case rules to apply to the caption
225 0352 1 CAPTION_IS_CENTERED  True if caption should be centered
226 0353 1 CAPTION_IS_FLUSH_RIGHT (not implemented) True if caption should be flush-right
227 0354 1 CAPTION_IS_RUN_IN    True if caption should be run-in with following text
228 0355 1 CAPTION_IS_BOLD      True if caption should be bolded
229 0356 1 CAPTION_IS_UNDERLINED True if caption should be underlined
230 0357 1 NEW_PAGE             start a new page before CHAPTER or APPENDIX.
231 0358 1                     Always TRUE for DSR.
232 0359 1 STARTODD            force the first page of a chapter to start on an odd page.
233 0360 1 TOCPAGE            insert a page number in the TOC.
234 0361 1 PUT_INTO_MEM_FILE   (not implemented) True if header should be output in .MEM file
235 0362 1 AUTOSUBTITLE       True if header should be picked up as an auto subtitle
236 0363 1 BRN_OPEN           True if header should be sent to the TOC
237 0364 1
238 0365 1 IMPLICIT INPUTS:    None
239 0366 1
240 0367 1 IMPLICIT OUTPUTS:  None
241 0368 1

```



```

242 0369 1 | ROUTINE VALUE:
243 0370 1 | COMPLETION CODES:      None
244 0371 1 |
245 0372 1 | SIDE EFFECTS:         None
246 0373 1 |
247 0374 1 |
248 0375 2 | BEGIN
249 0376 2 | LOCAL
250 0377 2 |     hold_khar,
251 0378 2 |     hold_lst_sp,
252 0379 2 |     hold_mra,
253 0380 2 |     hold_tsf,
254 0381 2 |     hold_lm,
255 0382 2 |     hold_wrd_pntr,
256 0383 2 |     ira_hold: VECTOR [10],
257 0384 2 |     hold_headers,                !Copy of HCT_HEADERS
258 0385 2 |     minor_code,
259 0386 2 |     sca_hold :sca_definition;
260 0387 2 |
261 0388 2 | !Report pending errors.
262 0389 2 | remneg ();
263 0390 2 | remmrg ();
264 0391 2 | tstblk (0);
265 0392 2 |
266 0393 2 | ! It's a chapter or appendix now.
267 0394 2 | npagen [sct_typ] = (IF .counter_minor_type EQL min_chapt_inf
268 0395 2 |     THEN sct_chapt
269 0396 2 |     ELSE sct_append);
270 0397 2 |
271 0398 2 | rskips (ira);                ! Skip spaces and tabs before the text.
272 0399 2 |
273 0400 2 | !We may have had chapters OR appendixes already if so begin
274 0401 2 | !counting where we left off.
275 0402 2 |
276 0403 2 | !It is possible that we are in the no_page mode and this new chapter
277 0404 2 | !will throw a page.  If the layout puts the page number at the bottom
278 0405 2 | !c. the page then we can't update the NPAGEN until we are at the next
279 0406 2 | !page.  Otherwise the page numbers come out incorrectly.
280 0407 2 | ! For the utilities, we must keep the sction numbering as it used to be.
281 0408 2 |
282 0409 2 | npagen [sct_number] = .counter_value;                !To maintain compatability
283 0410 2 | npagen [sct_page] = 1;                                !number the next page "1"
284 0411 2 | npagen [sct_sub_page] = 0;                            !turn off subpaging.
285 0412 2 | gca_chapt = true;                                     !Mark document as containing chapters/appendices.
286 0413 2 |
287 0414 2 |
288 0415 2 | ! Reset footnote number to 0.
289 0416 2 |
290 0417 2 | fnct_number_l = 0;
291 0418 2 | fnct_number_r = 0;
292 0419 2 | fnct_number   = 0;
293 0420 2 |
294 0421 2 | !See if this chapter is supposed to start on a new page.
295 U 0422 2 | %IF dsrplus %THEN
296 U 0423 2 | IF NOT .new_page
297 U 0424 2 | THEN
298 U 0425 2 | BEGIN

```

```

299 U 0426      gtpc (.test_page_amount);
300 U U 0427      outcrg ();          !Now, force the paper to be positioned as it should.
301 U U 0428      END
302 U U 0429      ELSE
303 U 0430      BEGIN
304 U 0431      %FI
305 U 0432      !Initialization of new chapter or appendix
306 U U 0433      %IF dsrplus %THEN
307 U U 0434      IF .startodd
308 U U 0435      THEN          !user wants to start on an odd page.
309 U U 0436      BEGIN
310 U U 0437      fs_wchar (mra, rintes);
311 U U 0438      fs_wchar (mra, %C'w');
312 U U 0439      fs_wchar (mra, %C' ');
313 U U 0440      tsf_int_vl = .tsf_int_vl + 3;
314 U U 0441      END
315 U 0442      ELSE
316 U 0443      %FI
317 U 0444      gcpage ();          !Start a new page
318 U U 0445      %IF dsrplus %THEN
319 U U 0446      !Turn off page headers for this page if not first title [always].
320 U U 0447      IF NOT .hct_title_always
321 U U 0448      THEN
322 U 0449      BEGIN
323 U 0450      %FI
324 U 0451      hold_headers = .hct_headers;          !No header at top of chapters and appendices.
325 U 0452      hct_headers = false;
326 U U 0453      %IF dsrplus %THEN
327 U 0454      END;
328 U 0455      %FI
329 U 0456      hct_odd_even = 0;          !Reset odd/even page parity.
330 U U 0457      %IF dsrplus %THEN
331 U 0458      END;
332 U 0459      %FI
333 U 0460
334 U 0461      IF .sca_autotitle          !If AUTOTITLE (default), then:
335 U 0462      THEN
336 U 0463      BEGIN          !Clear subtitles
337 U 0464      LOCAL
338 U 0465      hold_tsf;
339 U 0466
340 U 0467      hold_tsf = .tsf;          !Remember current TSF.
341 U 0468      tsf = sbttsf;          !Switch to the TSF for subtitles.
342 U 0469      tsf_int_hl = 0;
343 U 0470      tsf_ext_hl = 0;
344 U 0471      tsf = .hold_tsf;          !Return to main TSF.
345 U 0472      fs_init (sbtmra);          !Complete reset of subtitle MRA.
346 U 0473      END
347 U 0474      ELSE
348 U 0475      IF .gra_autosubt          !Auto subtitle is on, don't carry
349 U 0476      THEN          !over past subtitles.
350 U 0477      BEGIN          !Clear subtitles
351 U 0478      LOCAL
352 U 0479      hold_tsf;
353 U 0480
354 U 0481      hold_tsf = .tsf;          !Remember current TSF.
355 U 0482      tsf = sbttsf;          !Switch to the TSF for subtitles.

```

```

356      0483      3      tsf_int_hl = 0;
357      0484      3      tsf_ext_hl = 0;
358      0485      3      tsf = .hold tsf;          !Return to main TSF.
359      0486      3      fs_init (sbfmra);        !Complete reset of subtitle MRA.
360      0487      3      END;
361      0488
362      0489      3      !Generate the specified number of lines before the header text.
363      0490      3      !This is only done if we are starting the chapter/append on a new page.
364      0491      3      %IF dsrplus %THEN
365      0492      3      IF .new_page
366      0493      3      THEN
367      0494      3      BEGIN
368      0495      3
369      0496      3      !Skip 9 lines after the title, 8 after the subtitle, else skip 12.
370      0497      3      IF .hct_title_always
371      0498      3      THEN
372      0499      3      IF .hct_subtitle
373      0500      3      THEN guskip (.lines_before - 4)
374      0501      3      ELSE guskip (.lines_before - 3)
375      0502      3      ELSE
376      0503      3      guskip (.lines_before);
377      0504      3      END
378      0505      3      ELSE
379      0506      3      %FI
380      0507      3      guskip (.lines_before);          !end of top-of page-skip
381      0508
382      0509      3      !Reset header levels.
383      0510      3      INCR i FROM 1 TO .hllist [cl_max_index] DO
384      0511      3      hllist [.i] = 0;
385      0512
386      0513      3      hllist [cl_index] = 1;
387      0514
388      0515      3      !Reset entity counts for Example, Figure, Table.
389      0516      3      INCR i FROM 0 TO 2 DO
390      0517      3      ecc [.i, ecc$h_counter] = 0;
391      0518
392      0519      3      sdxy ();          !Reset some other things
393      0520
394      0521      3      IF .phan_top_first
395      0522      3      THEN
396      0523      3      !This was specified at the top of the very first page, and so
397      0524      3      !we cannot wait for NEWPAG to advance the page number.
398      0525      3      BEGIN
399      0526      3      pagen [sct_typ] = .npagen [sct_typ];
400      0527      3      pagen [sct_number] = .npagen [sct_number];
401      0528      3      pagen [sct_page] = .npagen [sct_page];
402      0529      3      npagen [sct_page] = .npagen [sct_page] + 1;
403      0530      3      END;
404      0531
405      0532      3      !If at the top of any page, output the page number. This is being done
406      0533      3      !because of problems with the the design of RUNOFF. NEWPAG won't output
407      0534      3      !the page number until this header text forces a title to be generated,
408      0535      3      !which may be too late in this case.
409      0536      3      IF NOT (.gca_op_dev EQL op_dev_flip)
410      0537      3      THEN
411      0538      3      BEGIN
412      0539      3      IF .brn_open

```

```

413 0540      THEN
414 0541          IF .phan_top_first
415 0542              THEN
416 0543                  puttpg (pagen, -1)
417 0544              ELSE
418 0545                  puttpg (npagen, -1);
419 0546      END;
420 0547
421 0548
422 0549      ! If creating a binary table of contents file, write out the counter and
423 0550      ! display descriptor.
424 0551      IF .brn_open
425 0552      THEN
426 0553          !User is generating a table of contents.
427 0554          putcnt ( .counter_major_type
428 0555                  ..counter_minor_type
429 0556                  ..tocpage
430 0557                  ,0
431 0558                  ..counter_pre_string_length
432 0559                  ..counter_pre_string_ptr
433 0560                  ..counter_post_string_length
434 0561                  ..counter_post_string_ptr
435 0562                  );
436 0563
437 0564      !Save scanning information
438 0565      push_sca;      !Save the SAVED SCA bits.
439 0566
440 0567      INCR i FROM 0 TO sca_size - 1 DO
441 0568          sca_hold [i] = .sca [i];
442 0569      hold_lm = .sca_lm;
443 0570
444 0571      INCR i FROM 0 TO 10 - 1 DO
445 0572          ira_hold [i] = .ira [i];
446 0573      hold_khar = .khar;
447 0574
448 0575
449 0576      !+
450 0577      ! If either (1) this head will become a title or (2) user is creating a
451 0578      ! .BRN file, then we must scan the text twice. The first time we scan it at full
452 0579      ! width, 150 characters, for title and/or TOC.
453 0580      ! If the title actually runs wider than the current right margin,
454 0581      ! however, we truncate it and append "...".
455 0582
456 0583      IF .sca_autotitle OR .brn_open
457 0584      THEN
458 0585          BEGIN
459 0586              sca_prescan = false;
460 0587              sca_rskips = true;
461 0588              sca_fill = false;
462 0589              sca_justify = false;
463 0590              !Set right margin big to catch long headers.
464 0591              sca_lm = 0;
465 0592              sca_rm = 150;
466 0593              sca_do_ind = false;
467 0594
468 0595              !Ignore indexing for subtitle/TOC scan.
469 0596
469 0596              !Make a title out of this chapter/appendix title, if that's what the
469 0596              ! user desires. The routine TITLES, which normally processes

```

```

: 470 0597 3 ! titles/subtitles, contains the necessary logic.
: 471 0598 3 IF .sca_autotitle
: 472 0599 3 THEN
: 473 0600 3 !Collect caption text in title buffer
: 474 0601 4 BEGIN
: 475 0602 4 !Set up bolding and underlining, if requested.
: 476 0603 5 IF (.caption_is_bold AND .sca_do_bld)
: 477 0604 4 THEN
: 478 0605 5 BEGIN
: 479 0606 5 sca_bld = true;
: 480 0607 5 sca_wrd_c_bld = true;
: 481 0608 4 END;
: 482 0609 4
: 483 0610 5 IF (.caption_is_underlined AND .sca_do_und)
: 484 0611 4 THEN
: 485 0612 5 BEGIN
: 486 0613 5 sca_und = true;
: 487 0614 5 sca_wrd_c_und = true;
: 488 0615 4 END;
: 489 0616 4
: 490 0617 4 setcas (.caption_case); !Set up case rules for the caption text.
: 491 0618 4 titles (h_title); !Use TITLES to get the title.
: 492 0619 3 END; !End of autotitle processing.
: 493 0620 3
: 494 0621 3 IF .brn_open
: 495 0622 3 THEN
: 496 0623 4 BEGIN
: 497 0624 4
: 498 0625 4 !Restore IRA and KHAR.
: 499 0626 4 INCR i FROM 0 TO 10 - 1 DO
: 500 0627 4 ira [.i] = .ira_hold [.i];
: 501 0628 4
: 502 0629 4 INCR i FROM 0 TO sca_size - 1 DO
: 503 0630 4 sca [.i] = .sca_hold [.i];
: 504 0631 4
: 505 0632 4 pop_sca; !Restore the SAVED SCA bits.
: 506 0633 4 !Save scanning information for another scan.
: 507 0634 4 push_sca; !Save the SAVED SCA bits.
: 508 0635 4
: 509 0636 4 INCR i FROM 0 TO sca_size - 1 DO
: 510 0637 4 sca_hold [.i] = .sca [.i];
: 511 0638 4
: 512 0639 4 !Save IRA and KHAR.
: 513 0640 4 INCR i FROM 0 TO 10 - 1 DO
: 514 0641 4 ira_hold [.i] = .ira [.i];
: 515 0642 4
: 516 0643 4 !Going to TOC, may or may not be an auto-title.
: 517 0644 4 sca_prescan = false; !A ';' does NOT terminate this command.
: 518 0645 4 sca_rskips = true; !Ignore multiple spaces and tabs.
: 519 0646 4 sca_fill = false;
: 520 0647 4 sca_justify = false;
: 521 0648 4 !Set right margin big to catch long headers.
: 522 0649 4 sca_lm = 0;
: 523 0650 4 sca_rm = 150;
: 524 0651 4 sca_do_ind = false; !Ignore indexing for title/TOC scan.
: 525 0652 4 sca_fc_case = true;
: 526 0653 4 sca_fc = true;

```

```

: 527      0654 4      khar = .hold_khar;
: 528      0655 4
: 529      0656 4      ! Switch to the Footnote TSF and MRA (which are not otherwise
: 530      0657 4      ! used in .ch/.ax) to collect the header information for the TOC.
: 531      0658 4
: 532      0659 4      hold_mra = .mra;
: 533      0660 4      mra = foomra;
: 534      0661 4      hold_tsf = .tsf;
: 535      0662 4      tsf = footsf;
: 536      0663 4
: 537      0664 4      !Before initializing the text descriptors, it is necessary to explicitly
: 538      0665 4      !reset the maximum length of the footnote MRA. It is clobbered if any
: 539      0666 4      !footnotes have been processed so far. See the comments in FONLY around
: 540      0667 4      !line 1545 for the detailed justification.
: 541      0668 4      fs_maxsize (mra) = s_fmra;
: 542      0669 4
: 543      0670 4      !Now initialize the text descriptors.
: 544      0671 4      fs_init (mra);
: 545      0672 4      INCR i FROM 0 TO tsf_size - 1 DO
: 546      0673 4      tsf [.i] = 0;
: 547      0674 4
: 548      0675 4      !Set up bolding and underlining, if requested.
: 549      0676 5      IF (.caption_is_bold AND .sca_do_bld)
: 550      0677 4      THEN
: 551      0678 5      BEGIN
: 552      0679 5      sca_bld = true;
: 553      0680 5      sca_wrd_c_bld = true;
: 554      0681 4      END;
: 555      0682 4
: 556      0683 5      IF (.caption_is_underlined AND .sca_do_und)
: 557      0684 4      THEN
: 558      0685 5      BEGIN
: 559      0686 5      sca_und = true;
: 560      0687 5      sca_wrd_c_und = true;
: 561      0688 4      END;
: 562      0689 4
: 563      0690 4      !Set up case rules for the chapter/appendix.
: 564      0691 4      setcas (.caption_case);
: 565      0692 4
: 566      0693 4      !Scan 150-wide into footnote MRA.
: 567      0694 4      scant ();
: 568      0695 4      endwrd (false, false, false);
: 569      0696 4
: 570      0697 4      !Switch TSF and MRA back.
: 571      0698 4      mra = .hold_mra;
: 572      0699 4      tsf = .hold_tsf;
: 573      0700 3      END;
: 574      0701 3
: 575      0702 3      !Restore IRA and KHAR.
: 576      0703 3      INCR i FROM 0 TO 10 - 1 DO
: 577      0704 3      ira [.i] = .ira_hold [.i];
: 578      0705 3
: 579      0706 3      khar = .hold_khar;
: 580      0707 2      END;
: 581      0708 2
: 582      0709 2      INCR i FROM 0 TO sca_size - 1 DO
: 583      0710 2      sca [.i] = .sca_hold [.i];

```

!End of title-or-TOC processing

.....

```

: 584 0711 2
: 585 0712 2
: 586 0713 2
: 587 0714 2
: 588 0715 2
: 589 0716 2
: 590 0717 2
: 591 0718 2
: 592 0719 2
: 593 0720 2
: 594 0721 2
: 595 0722 2
: 596 0723 2
: 597 0724 2
: 598 0725 2
: 599 0726 2
: 600 0727 2
: 601 0728 2
: 602 0729 2
: 603 0730 2
: 604 0731 2
: 605 0732 2
: 606 0733 2
: 607 0734 2
: 608 0735 2
: 609 0736 2
: 610 0737 2
: 611 0738 2
: 612 0739 2
: 613 0740 2
: 614 0741 2
: 615 0742 2
: 616 0743 3
: 617 0744 4
: 618 0745 4
: 619 0746 4
: 620 0747 4
: 621 0748 4
: 622 0749 4
: 623 0750 3
: 624 0751 2
: 625 0752 2
: 626 0753 2
: 627 0754 2
: 628 0755 2
: 629 0756 2
: 630 0757 3
: 631 0758 4
: 632 0759 3
: 633 0760 4
: 634 0761 4
: 635 0762 4
: 636 0763 4
: 637 0764 4
: 638 0765 4
: 639 0766 4
: 640 0767 4

pop_sca;      !Restore the SAVED SCA bits.
sca_fill = true;      !Fill the header even if not filling text.
sca_prescan = false;  !A ';' does NOT terminate this command.
sca_rskips = true;    !Ignore multiple spaces and tabs.

!Set up bolding and underlining, if requested.
IF (.caption_is_bold AND .sca_do_bld)
THEN
BEGIN
sca_bld = true;
sca_wrd_c_bld = true;
END;

IF (.caption_is_underlined AND .sca_do_und)
THEN
BEGIN
sca_und = true;
sca_wrd_c_und = true;
END;

! Generate the number for the .ch/.ax if not turned off.
IF show_counter
THEN
!User didn't turn off the numbering, so generate the .ch/.ax counter.
BEGIN
!Generate the counter in using the case rules for the caption.
outctr (.counter_minor_type, .caption_case);

!Put the counter into the output buffer.
fs_next (fs01) = .fs_start (fs01);

INCR i FROM 1 TO .fs_length (fs01) DO
BEGIN
LOCAL
temp_char;

fs_rchar (fs01, temp_char);
endchr (.temp_char);
END;

! End the string so the spaces_after will not get underlined (if in
! effect).
endwrd (false, false, false);

!Insert spaces after the counter if user didn't say BREAK or BETWEEN.
!Do not put out more than 75 spaces regardless of what the user said.
IF NOT (.break_before_caption OR (.lines_between GTR 0))
THEN
BEGIN
LOCAL
sca_hold_c_bldun,
sca_hold_ac_blun;

!ENDCHR seems to carry over underlining rules even for spaces.
!We must turn off these bits before we add the extra spaces.
sca_hold_c_bldun = .sca_wrd_c_bldun;

```

```

641 0768 4          sca_hold_ac_blun = .sca_wrd_ac_blun;
642 0769 4
643 0770 4          INCR i FROM 1 TO (MIN (.counter_spaces_after, 75)) DO
644 0771 5          BEGIN
645 0772 5          sca_wrd_c_bldun = 0;
646 0773 5          sca_wrd_ac_blun = 0;
647 0774 5          endchr ('%C^ ');
648 0775 4          END;
649 0776 4
650 0777 4          !Restore the SCA bits.
651 0778 4          sca_wrd_c_bldun = .sca_hold_c_bldun;
652 0779 4          sca_wrd_ac_blun = .sca_hold_ac_blun;
653 0780 4          END;
654 0781 3
655 0782 3          !And now end this 'word'.
656 0783 3          endwrd (false,false,false);
657 0784 3
658 0785 3          !If the user wants a break before the caption, we must
659 0786 3          !make any adjustments if not flush_left.
660 0787 3          IF .caption_is_centered OR .caption_is_flush_right
661 0788 3          THEN
662 0789 4          BEGIN
663 0790 4
664 0791 5          IF .break_before_caption OR (.lines_between GTR 0)
665 0792 4          THEN
666 0793 5          BEGIN
667 0794 5          tsf_adjust =
668 0795 6          (IF .caption_is_centered          !Center the counter.
669 0796 6          THEN (.sca_rm = (.tsf_ext_hl))/2 !Ignore spaces_after
670 0797 6          ELSE                               !Force the counter right.
671 0798 5          .sca_rm - .tsf_ext_hl);
672 0799 5          END
673 0800 4          ELSE
674 0801 5          BEGIN
675 0802 5          !No break to be done so we must set the left margin so wrapping
676 0803 5          !will left justify on the caption left margin.
677 0804 5          hold_lm = .tsf_ext_hl;          !Remember for use after SCA is restored.
678 0805 5          sca_lm = .tsf_ext_hl;
679 0806 4          END;
680 0807 4          END
681 0808 4
682 0809 4          !+
683 0810 4          ! The header is flush-left. If the caption is to immediately follow,
684 0811 4          ! set the left margin so that following text will wrap properly.
685 0812 4          !-
686 0813 3          ELSE
687 0814 3          !If we aren't doing a break after the counter, we must pull in
688 0815 3          !the left margin to prevent writing over the counter.
689 0816 4          IF NOT (.break_before_caption OR (.lines_between GTR 0))
690 0817 3          THEN
691 0818 3          !Set left margin beyond the counter so if the caption wraps,
692 0819 3          !it will be nicely justified with itself.
693 0820 3          sca_lm = .tsf_ext_hl;
694 0821 3
695 0822 3          END
696 0823 2          ELSE
697 0824 2          !

```



```

698 0825 2      !If we are not generating a number, then the first character
699 0826 2      !scanned should be the first character in this MRA.
700 0827 2
701 0828 2      sca_fc = true;
702 0829 2
703 0830 2      !+ If the header is other than flush-left and user asked for break or
704 0831 2      !some lines between, we have already computed amount to shift it,
705 0832 2      !we now put it out on a line by itself.
706 0833 2
707 0834 2      gca_line_pend = 1;          !Until we throw the header, a line is pending.
708 0835 2
709 0836 2      IF (.break_before_caption) OR (.lines_between GTR 0)
710 0837 2      THEN
711 0838 2          BEGIN
712 0839 2              outnj ();          !Put out the counter.
713 0840 2              gcskip(.lines_between); !Skip the number of lines requested.
714 0841 2          END;
715 0842 2
716 0843 2      ! Write text to .BRN file. The call to PUTTXT is done here, between
717 0844 2      ! generating the counter and the caption, so that coordination with
718 0845 2      ! FLIP/BIND is maintained.
719 0846 2
720 0847 2      IF .brn_open
721 0848 2      THEN
722 0849 2          ! Write to .BRN (or .BFL) file.
723 0850 2
724 0851 2          puttxt (.fs_length (foomra), .fs_start (foomra),
725 0852 2              .caption_major_type, .caption_minor_type);
726 0853 2
727 0854 2      !Set up case ruler for the heading.
728 0855 2      setcas (.caption_case);
729 0856 2
730 0857 2      !Tell SCANT that the first character of the header is the first
731 0858 2      !character of a word. (ENDCHR turned this stuff off before.)
732 0859 2      sca_fc_case = true;
733 0860 2
734 0861 2      scant ();          !Go get the caption.
735 0862 2
736 0863 2      !SCA_WRD_CPEND equals rintex IFF there was a trailing space/tab
737 0864 2      !after the text. In such a case ENDWRD has already been called.
738 0865 2      !Calling it again would have the effect of forcing an additional
739 0866 2      !space out into TSF/MRA.
740 0867 2      IF .sca_wrd_cpend NEQ rintex
741 0868 2      THEN
742 0869 2          endwrld (false, false, false)
743 0870 2      ELSE
744 0871 2
745 0872 2          IF .sca_wrd_lst_und EQL 0
746 0873 2              AND
747 0874 2              .sca_wrd_lst_sp GTR 0
748 0875 2          THEN
749 0876 2              !Chop off trailing spaces/tabs. When doing so, also back up
750 0877 2              !intra-line pointer and counter appropriately. If justification
751 0878 2              !was in effect, also cancel the justification mark that got writ-
752 0879 2              !ten onto the MRA.
753 0880 2              !Note that trailing spaces that are underlined are not discarded.
754 0881 2          BEGIN

```

```

755 0882 LOCAL
756 0883     chars_to_drop;
757 0884
758 0885     chars_to_drop = .sca_wrd_lst_sp +
759 0886     ( IF .sca_justify
760 0887     THEN 3
761 0888     ELSE 0
762 0889     );
763 0890     fs_length (mra) = .fs_length (mra) - .chars_to_drop;
764 0891     fs_next (mra) = CH$PLOS (.fs_next (mra), -.chars_to_drop);
765 0892     sca_wrd_lst_sp = 0;
766 0893 END;
767 0894
768 0895 !Restore previous scanner rules and set standard rules.
769 0896 hold_wrd_pntr = .sca_wrd_pntr;           !Remember start of next word.
770 0897 hold_lst_sp = .sca_wrd_lst_sp;         !Remember last-space info.
771 0898
772 0899 INCR i FROM 0 TO sca_size - 1 DO
773 0900     sca [.i] = .sca_hold [.i];
774 0901
775 0902 pop_sca;           !Restore the SAVED SCA bits again.
776 0903 sca_wrd_pntr = .hold_wrd_pntr;         !Restore start of next word.
777 0904 sca_wrd_lst_sp = .hold_lst_sp;       !Restore last-space info.
778 0905
779 0906 !This section basically outputs the last line of the header. This
780 0907 !may also be the first (if only one line long). What must happen here
781 0908 !is if the header is flush-left, the last line (if different from the first,
782 0909 !must start at the caption left margin. The same goes for centered
783 0910 !captions if nobreak is in effect and if the caption is so long that
784 0911 !it wraps. Now if it doesn't wrap, then this section will output
785 0912 !the whole counter and caption. By using GCA_LINE_PEND, we may determine
786 0913 !if this is the first line output. If a line has already gone out, then
787 0914 !SCANT has already adjusted the left margin and no TSF_adjust is needed.
788 0915
789 0916 !If the header is other than flush-left, compute amount to shift it.
790 0917 !If user asked for a BREAK or had some LINES_BETWEEN, then the shift
791 0918 !for the counter has already been taken care of and this section
792 0919 !is only for the caption, otherwise the shift is for the whole header.
793 0920 IF .caption_is_centered OR .caption_is_flush_right
794 0921 THEN
795 0922
796 0923     IF .gca_line_pend           !NO line thrown yet.
797 0924     THEN
798 0925         tsf_adjust =
799 0926         (IF .caption_is_centered           !Center the counter/and/or caption.
800 0927         THEN (.sca_rm - .tsf_ext_hl)/2
801 0928         ELSE .sca_rm - .tsf_ext_fl )       !Force the counter/caption right.
802 0929     ELSE
803 0930         IF (.break_before_caption) OR (.lines_between GTR 0)
804 0931         THEN
805 0932             tsf_adjust =
806 0933             (IF .caption_is_centered           !Center the counter/and/or caption.
807 0934             THEN (.sca_rm - .tsf_ext_hl)/2
808 0935             ELSE .sca_rm - .tsf_ext_fl);    !Force the counter/caption right.
809 0936
810 0937
811 0938 outnj ();           !force out the caption.

```

```

: 812      0939      2
: 813      0940      2      gcskip (.lines_after);
: 814      0941      2
: 815      0942      2      sca_sect_empty = true;          !This section is empty now.
: 816      0943      2
: 817      0944      2      !Set proper case conversion for remainder of section.
: 818      0945      2      setcas (.gca_case);
: 819      0946      2
: 820      0947      2      gca_pchax = false;          !Pending .NUMBER CHAPTER/APPENDIX done.
: 821      0948      2
: 822      U 0949      2      %IF dsrplus %THEN
: 823      U 0950      2      IF NOT .hct_title_always AND .new_page
: 824      U 0951      2      THEN
: 825      0952      2      %FI
: 826      0953      2      hct_headers = .hold_headers;          !Restore .HEADERS or .NO HEADERS status.
: 827      0954      2
: 828      U 0955      2      %IF dsrplus %THEN
: 829      U 0956      2      !We know now that the chapter counter has been output.
: 830      U 0957      2      !If we have moved to the top of a page fine, but if not,
: 831      U 0958      2      !we must update the PAGEN structure. We update if we did not explicitly
: 832      U 0959      2      !request a new page.
: 833      U 0960      2      IF NOT .new_page
: 834      U 0961      2      THEN
: 835      U 0962      2      BEGIN
: 836      U 0963      2      pagen [sct_number] = .counter_value;
: 837      U 0964      2      pagen [sct_typ] = (IF .counter_minor_type EQL min_chapt_inf
: 838      U 0965      2      THEN sct_chapt
: 839      U 0966      2      ELSE sct_append);
: 840      U 0967      2      npagen [sct_page] = .npagen [sct_page] + 1;
: 841      U 0968      2      END;
: 842      0969      2      %FI
: 843      0970      1      END;          !End of OUTCHA

```

.TITLE OUTCHA Processor for chapter and appendix headers.

.IDENT \V04-000\

.PSECT \$OWNS,NOEXE,2

00000 PP_SCA: .BLKB 48

- .EXTRN RINTES, S FMRA, ECC
- .EXTRN FNCT, FOMRA, FOOTSF
- .EXTRN FS01, GCA, HCT, HLDSP
- .EXTRN HLLIST, IRA, MRA
- .EXTRN TITMRA, TITTSF, NPAGEN
- .EXTRN PAGEN, PHAN, SBTMRA
- .EXTRN SBTTSF, SCA, TSF
- .EXTRN KHAR, ENDCHR, ENDWRD
- .EXTRN GCPAGE, GCSKIP, GTPC
- .EXTRN GUSKIP, OUTCRG, OUTCTR
- .EXTRN OUTNJ, PACSEC, PACXXX
- .EXTRN PUTCNT, PUTPG, PUTTXT
- .EXTRN REMMRG, REMNEG, RSKIPS
- .EXTRN SCANT, SDXY, SETCAS
- .EXTRN TITLES, TSTBLK

				.PSECT	SCODES,NOWRT,2		
				.ENTRY	OUTCHA, Save R2,R3,R4,R5,R6,R7,R8,R9,R10,-	:	
			OFFC 00000		R11	:	0313
		5B	00000000G EF 9E 00002	MOVAB	TSF, R11		
		5A	00000000' EF 9E 00009	MOVAB	PP SCA, R10		
		59	00000000G EF 9F 0001C	MOVAB	SCA+120, R9		
		5E	FE58 CE 9E 00017	MOVAB	-424(SP), SP		
	00000000G	EF	00 FB 0001C	CALLS	#0, REMNEG	:	0389
	00000000G	EF	00 FB 00023	CALLS	#0, REMMRG	:	0390
			7E D4 0002A	CLRL	-(SP)	:	0391
	00000000G	EF	01 FB 0002C	CALLS	#1, TSTBLK		
		10	14 AC D1 00033	CMPL	COUNTER_MINOR_TYPE, #16	:	0394
			05 12 00037	BNEQ	1\$		
		50	01 D0 00039	MOVL	#1, R0		
			03 11 0003C	BRB	2\$		
		50	03 D0 0003E 1\$:	MOVL	#3, R0		
00000000G	EF	00	50 F0 00041 2\$:	INSV	R0, #0, #4, NPAGEN		
			00000000G EF 9F 0004A	PUSHAB	IRA	:	0398
	00000000G	EF	01 FB 00050	CALLS	#1, RSKIPS		
	00000000G	EF	18 AC D0 00057	MOVL	COUNTER VALUE, NPAGEN+4	:	0409
	00000000G	EF	01 D0 0005F	MOVL	#1, NPAGEN+8	:	0410
	00000000G	EF	00000000G EF B4 00066	CLRW	NPAGEN+2	:	0411
			00000000G EF 01 D0 0006C	MOVL	#1, GCA+40	:	0412
			00000000G EF 7C 00073	CLRQ	FNCT+36	:	0417
			00000000G EF D4 00079	CLRL	FNCT+44	:	0419
	00000000G	EF	00 FB 0007F	CALLS	#0, GCPAGE	:	0444
		58	00000000G FF D0 00086	MOVL	@HCT+8, HOLD_HEADERS	:	0451
			00000000G FF D4 0008D	CLRL	@HCT+8	:	0452
			00000000G EF D4 00093	CLRL	HCT+16	:	0456
		07	14 B9 E8 00099	BLBS	@SCA+140, 3\$:	0461
		2E	00000000G FF E9 0009D	BLBC	@GCA+8, 4\$:	0475
		51	6B D0 000A4 3\$:	MOVL	TSF, HOLD TSF	:	0481
		6B	00000000G EF 9E 000A7	MOVAB	SBTTSF, TSF	:	0482
		50	6B D0 000AE	MOVL	TSF, R0		
			60 7C 000B1	CLRQ	(R0)	:	0483
		6B	51 D0 000B3	MOVL	HOLD TSF, TSF	:	0485
			00000000G EF D4 000B6	CLRL	SBTMRA+12	:	0486
	00000000G	EF	00000000G EF 9E 000BC	MOVAB	SBTMRA+16, SBTMRA		
	00000000G	EF	00000000G EF D0 000C7	MOVL	SBTMRA, SBTMRA+4		
			04 AC DD 000D2 4\$:	PUSHL	LINES BEFORE	:	0507
	00000000G	EF	01 FB 000D5	CALLS	#1, GOSKIP		
		51	00000000G EF D0 000DC	MOVL	HLLIST, R1	:	0510
			50 D4 000E3	CLRL	I		
			07 11 000E5	BRB	6\$		
			00000000GEF40 D4 000E7 5\$:	CLRL	HLLIST+4[I]	:	0511
	F5	50	51 F3 000EE 6\$:	AOBLEQ	R1, I, 5\$		
	00000000G	EF	01 D0 000F2	MOVL	#1, HLLIST+4	:	0513
			50 D4 000F9	CLRL	I	:	0516
	51	50	24 C5 000FB 7\$:	MULL3	#36, I, R1	:	0517
00000000GEF41	20	18	00 F0 000FF	INSV	#0, #24, #32, ECC+12[R1]		
	EE	50	02 F3 00109	AOBLEQ	#2, I, 7\$		
			00000000G EF 00 FB 0010D	CALLS	#0, SDXY	:	0519
		1E	00000000G EF E9 00114	BLBC	PHAN+24, 8\$:	0521
	00000000G	EF	00 00000000G EF F0 0011B	INSV	NPAGEN, #0, #4, PAGEN	:	0526
			00000000G EF 7D 00128	MOVQ	NPAGEN+4, PAGEN+4	:	0527

02	00000000G	EF	04	00000000G	EF	D6	00133		INCL	NPAGEN+8	0529
					04	ED	00139	8\$:	CMPZV	#4, #4, GCA+208, #2	0536
					26	13	00142		BEQL	11\$	
			3E	5C	AC	E9	00144		BLBC	BRN_OPEN, 12\$	0539
			0B	00000000G	EF	E9	00148		BLBC	PHAR+24, 9\$	0541
			7E		01	CE	0014F		MNEGL	#1, -(SP)	0543
					00000000G	EF	9F	00152	PUSHAB	PAGEN	
					09	11	00158		BRB	10\$	
			7E		01	CE	0015A	9\$:	MNEGL	#1, -(SP)	0545
					00000000G	EF	9F	0015D	PUSHAB	NPAGEN	
	00000000G		EF		02	FB	00163	10\$:	CALLS	#2, PUTTPG	
			18	5C	AC	E9	0016A	11\$:	BLBC	BRN_OPEN, 12\$	0551
			7E	2C	AC	7D	0016E		MOVQ	COUNTER_POST_STRING_LENGTH, -(SP)	0560
			7E	24	AC	7D	00172		MOVQ	COUNTER_PRE_STRING_LENGTH, -(SP)	0558
					7E	D4	00176		CLRL	-(SP)	0554
					70	AC	00178		PUSHL	TOCPAGE	0556
			7E	10	AC	7D	0017B		MOVQ	COUNTER_MAJOR_TYPE, -(SP)	0554
	00000000G		EF		08	FB	0017F		CALLS	#8, PUTCNT	
			04	AA	EC	B9	00186	12\$:	MOVL	@SCA+100, PP_SCA	0562
			08	AA	F0	B9	0018A		MOVL	@SCA+104, PP_SCA+4	
			0C	AA	F4	B9	0018F		MOVL	@SCA+108, PP_SCA+8	
			10	AA	F8	B9	00194		MOVL	@SCA+112, PP_SCA+12	
			14	AA	FC	B9	00199		MOVL	@SCA+116, PP_SCA+16	
			18	AA	00	B9	0019E		MOVL	@SCA+120, PP_SCA+20	
			1C	AA	04	B9	001A3		MOVL	@SCA+124, PP_SCA+24	
			20	AA	08	B9	001A8		MOVL	@SCA+128, PP_SCA+28	
			24	AA	0C	B9	001AD		MOVL	@SCA+132, PP_SCA+32	
			28	AA	10	B9	001B2		MOVL	@SCA+136, PP_SCA+36	
			2C	AA	14	B9	001B7		MOVL	@SCA+140, PP_SCA+40	
					18	B9	001BC		MOVL	@SCA+144, PP_SCA+44	
					50	D4	001C1		CLRL	I	0567
			6E40		88	A940	001C3	13\$:	MOVL	SCA[I], SCA_HOLD[I]	0568
F2			50	0000005F	8F	F3	001C9		AOBLEQ	#95, I, 13\$	
			57		FC	B9	001D1		MOVL	@SCA+116, HOLD_LM	0569
					50	D4	001D5		CLRL	I	0571
F2	D8	AD40	00000000G	EF	40	D0	001D7	14\$:	MOVL	IRA-4[I], IRA_HOLD[I]	0572
			50		09	F3	001E1		AOBLEQ	#9, I, 14\$	
			54	00000000G	EF	D0	001E5		MOVL	KHAR, HOLD_KHAR	0573
			07		14	B9	001EC		BLBS	@SCA+140, 15\$	0582
			03		5C	AC	001F0		BLBS	BRN_OPEN, 15\$	
					01D6	31	001F4		BRW	29\$	
			54	A9	34	A9	001F7	15\$:	CLRL	SCA+172	0586
					01	D0	001FA		MOVL	#1, SCA+204	0587
					F0	B9	001FE		CLRL	@SCA+104	0588
					EC	B9	00201		CLRL	@SCA+100	0589
					FC	B9	00204		CLRL	@SCA+116	0591
			00	B9	96	8F	9A	00207	MOVZBL	#150, @SCA+120	0592
			30	A9	08	8A	0020C		BICB2	#8, SCA+168	0593
					14	B9	00210		BLBC	@SCA+140, 18\$	0598
					4C	AC	00214		BLBC	CAPTION_IS_BOLD, 16\$	0603
					08	A9	00218		BLBC	SCA+168, 16\$	
			20	A9	01	88	0021C		BISB2	#1, SCA+152	0606
			4C	A9	01	88	00220		BISB2	#1, SCA+196	0607
					50	AC	00224	16\$:	BLBC	CAPTION_IS_UNDERLINED, 17\$	0610
08			30	A9	01	E1	00228		BBC	#1, SCA+168, 17\$	
			20	A9	02	88	0022D		BISB2	#2, SCA+152	0613
			4C	A9	02	88	00231		BISB2	#2, SCA+196	0614

			3C	AC	DD	00235	17\$:	PUSHL	CAPTION_CASE		0617
	00000000G	EF		01	FB	00238		CALLS	#1, SETCAS		
		7E	CC	8F	9A	0023F		MOVZBL	#204, -(SP)		0618
	00000000G	EF		01	FB	00243		CALLS	#1, TITLES		
		03	5C	AC	E8	0024A	18\$:	BLBS	BRN_OPEN, 19\$		0621
				0165	31	0024E		BRW	27\$		
				50	D4	00251	19\$:	CLRL	I		0626
	00000000GEF40		D8	AD40	D0	00253	20\$:	MOVL	IRA_HOLD[I], IRA-4[I]		0627
F2		50		09	F3	0025D		AOBLEQ	#9, I, 20\$		
				50	D4	00261		CLRL	I		0629
	88 A940		6E40	D0	00263		21\$:	MOVL	SCA_HOLD[I], SCA[I]		0630
F2		50	0000005F	8F	F3	00269		AOBLEQ	#95, I, 21\$		
	EC	B9		6A	D0	00271		MOVL	PP_SCA, @SCA+100		
	FO	B9	04	AA	D0	00275		MOVL	PP_SCA+4, @SCA+104		
	F4	B9	08	AA	D0	0027A		MOVL	PP_SCA+8, @SCA+108		
	F8	B9	0C	AA	D0	0027F		MOVL	PP_SCA+12, @SCA+112		
	FC	B9	10	AA	D0	00284		MOVL	PP_SCA+16, @SCA+116		
	00	B9	14	AA	D0	00289		MOVL	PP_SCA+20, @SCA+120		
	04	B9	18	AA	D0	0028E		MOVL	PP_SCA+24, @SCA+124		
	08	B9	1C	AA	D0	00293		MOVL	PP_SCA+28, @SCA+128		
	0C	B9	20	AA	D0	00298		MOVL	PP_SCA+32, @SCA+132		
	10	B9	24	AA	D0	0029D		MOVL	PP_SCA+36, @SCA+136		
	14	B9	28	AA	D0	002A2		MOVL	PP_SCA+40, @SCA+140		
	18	B9	2C	AA	D0	002A7		MOVL	PP_SCA+44, @SCA+144		
		6A	EC	B9	D0	002AC		MOVL	@SCA+100, PP_SCA		0632
	04	AA	FO	B9	D0	002B0		MOVL	@SCA+104, PP_SCA+4		
	08	AA	F4	B9	D0	002B5		MOVL	@SCA+108, PP_SCA+8		
	0C	AA	F8	B9	D0	002BA		MOVL	@SCA+112, PP_SCA+12		
	10	AA	FC	B9	D0	002BF		MOVL	@SCA+116, PP_SCA+16		
	14	AA	00	B9	D0	002C4		MOVL	@SCA+120, PP_SCA+20		
	18	AA	04	B9	D0	002C9		MOVL	@SCA+124, PP_SCA+24		
	1C	AA	08	B9	D0	002CE		MOVL	@SCA+128, PP_SCA+28		
	20	AA	0C	B9	D0	002D3		MOVL	@SCA+132, PP_SCA+32		
	24	AA	10	B9	D0	002D8		MOVL	@SCA+136, PP_SCA+36		
	28	AA	14	B9	D0	002DD		MOVL	@SCA+140, PP_SCA+40		
	2C	AA	18	B9	D0	002E2		MOVL	@SCA+144, PP_SCA+44		
				50	D4	002E7		CLRL	I		0636
	6E40		88 A940	D0	002E9		22\$:	MOVL	SCA[I], SCA_HOLD[I]		0637
F2		50	0000005F	8F	F3	002EF		AOBLEQ	#95, I, 22\$		
				50	D4	002F7		CLRL	I		0640
	D8	AD40	00000000GEF40	D0	002F9		23\$:	MOVL	IRA-4[I], IRA_HOLD[I]		0641
F2		50		09	F3	00303		AOBLEQ	#9, I, 23\$		
				34	A9	D4	00307	CLRL	SCA+172		0644
	54	A9		01	D0	0030A		MOVL	#1, SCA+204		0645
				FO	B9	D4	0030E	CLRL	@SCA+104		0646
				EC	B9	D4	00311	CLRL	@SCA+100		0647
				FC	B9	D4	00314	CLRL	@SCA+116		0648
	00	B9		96	8F	9A	00317	MOVZBL	#150, @SCA+120		0649
	30	A9		08	8A	0031C		B!CB2	#8, SCA+168		0650
	58	A9		01	D0	00320		MOVL	#1, SCA+208		0651
	1C	A9		01	D0	00324		MOVL	#1, SCA+148		0652
	00000000G	EF		54	D0	00328		MOVL	HOLD_KHAR, KHAR		0653
		53	00000000G	EF	D0	0032F		MOVL	MRA_HOLD MRA		0654
	U0000000G	EF	00000000G	EF	9E	00336		MOVAB	FOOMRA, MRA		0655
		52		6B	D0	00341		MOVL	TSF, HOLD TSF		0656
		68	00000000G	EF	9E	00344		MOVAB	FOO!SF, TSF		0657
		50	00000000G	EF	D0	00348		MOVL	MRA, RO		0658

	08	A0	00000000G	8F	D0	00352	MOVL	#S FMRA, 8(R0)		
				0C	A0	D4 0035A	CLRL	12(R0)		0671
		60		10	A0	9E 0035D	MOVAB	16(R0), (R0)		
	04	A0			60	D0 00361	MOVL	(R0), 4(R0)		
					50	D4 00365	CLRL	I		0673
			00	BB40	D4	00367	CLRL	@TSF[I]		
F8		50		27	F3	00368	AOBLEQ	#39, I, 24\$		
		0C	4C	AC	E9	0036F	BLBC	CAPTION IS BOLD, 25\$		0676
		08	30	A9	E9	00373	BLBC	SCA+168, 25\$		
	20	A9		01	88	00377	BISB2	#1, SCA+152		0679
	4C	A9		01	88	0037B	BISB2	#1, SCA+196		0680
		0D	50	AC	E9	0037F	BLBC	CAPTION IS UNDERLINED, 26\$		0683
08		30		01	E1	00383	BBC	#1, SCA+168, 26\$		
	20	A9		02	88	00388	BISB2	#2, SCA+152		0686
	4C	A9		02	88	0038C	BISB2	#2, SCA+196		0687
			3C	AC	DD	00390	PUSHL	CAPTION CASE		0691
	00000000G	EF		01	FB	00393	CALLS	#1, SETCAS		
	00000000G	EF		00	FB	0039A	CALLS	#0, SCANT		0694
				7E	7C	003A1	CLRQ	-(SP)		0695
				7E	D4	003A3	CLRL	-(SP)		
	00000000G	EF		03	FB	003A5	CALLS	#3, ENDWRD		0698
	00000000G	EF		53	D0	003AC	MOVL	HOLD_MRA, MRA		0699
		6B		52	D0	003B3	MOVL	HOLD_TSF, TSF		0703
				50	D4	003B6	CLRL	I		0704
	00000000G	EF	40	DB	AD40	D0 003B8	MOVL	IRA_HOLD[I], IRA-4[I]		
F2		50		09	F3	003C2	AOBLEQ	#9, I, 28\$		
	00000000G	EF		54	D0	003C6	MOVL	HOLD_KHAR, KHAR		0706
				50	D4	003CD	CLRL	I		0709
		88	A940	6E40	D0	003CF	MOVL	SCA_HOLD[I], SCA[I]		0710
F2		50	0000005F	8F	F3	003D5	AOBLEQ	#95, I, 30\$		
	EC	B9		6A	D0	003D9	MOVL	PP_SCA, @SCA+100		
	FO	B9	04	AA	D0	003E1	MOVL	PP_SCA+4, @SCA+104		
	F4	B9	08	AA	D0	003E6	MOVL	PP_SCA+8, @SCA+108		
	F8	B9	0C	AA	D0	003EB	MOVL	PP_SCA+12, @SCA+112		
	FC	B9	10	AA	D0	003F0	MOVL	PP_SCA+16, @SCA+116		
	00	B9	14	AA	D0	003F5	MOVL	PP_SCA+20, @SCA+120		
	04	B9	18	AA	D0	003FA	MOVL	PP_SCA+24, @SCA+124		
	08	B9	1C	AA	D0	003FF	MOVL	PP_SCA+28, @SCA+128		
	0C	B9	20	AA	D0	00404	MOVL	PP_SCA+32, @SCA+132		
	10	B9	24	AA	D0	00409	MOVL	PP_SCA+36, @SCA+136		
	14	B9	28	AA	D0	0040E	MOVL	PP_SCA+40, @SCA+140		
	18	B9	2C	AA	D0	00413	MOVL	PP_SCA+44, @SCA+144		
	FO	B9		01	D0	00418	MOVL	#1, @SCA+104		0713
			34	A9	D4	0041C	CLRL	SCA+172		0714
	54	A9		01	D0	0041F	MOVL	#1, SCA+204		0715
		0C	4C	AC	E9	00423	BLBC	CAPTION IS BOLD, 31\$		0718
		08	30	A9	E9	00427	BLBC	SCA+168, 31\$		
	20	A9		01	88	0042B	BISB2	#1, SCA+152		0721
	4C	A9		01	88	0042F	BISB2	#1, SCA+196		0722
		0D	50	AC	E9	00433	BLBC	CAPTION IS UNDERLINED, 32\$		0723
08		30		01	E1	00437	BBC	#1, SCA+168, 32\$		
	20	A9		02	88	0043C	BISB2	#2, SCA+152		0728
	4C	A9		02	88	00440	BISB2	#2, SCA+196		0729
		38		01	7A	00444	EMUL	#1, CAPTION_MINOR_TYPE, #0, -(SP)		0733
7E	00			03	7B	0044A	EDIV	#3, (SP)+, R0, R0		
50	50			50	D1	0044F	CMPL	R0, #2		
				0F	13	00452	BEQL	33\$		

			03	38	AC	D1	00454		C MPL	CAPTION_MINOR_TYPE, #3			
					09	13	00458		BEQL	33\$			
			04	38	AC	D1	0045A		C MPL	CAPTION_MINOR_TYPE, #4			
					03	13	0045E		BEQL	33\$			
					00F0	31	00460		BRW	48\$			
					3C	AC	DD	00463	33\$:	PUSHL	CAPTION_CASE	0738	
					14	AC	DD	00466		PUSHL	COUNTER_MINOR_TYPE		
		00000000G	EF		02	FB	00469		CALLS	#2, OUTCTR			
		00000000G	EF	00000000G	EF	D0	00470		MOVL	FS01, FS01+4		0741	
			53	00000000G	EF	D0	0047B		MOVL	FS01+12, R3		0743	
					52	D4	00482		CLRL	I			
					1C	11	00484		BRB	35\$			
			50	00000000G	FF	9A	00486	34\$:	MOVZBL	@FS01+4, TEMP_CHAR		0748	
				00000000G	EF	D6	0048D		INCL	FS01+4			
				00000000G	EF	D7	00493		DECL	FS01+12			
					50	DD	00499		PUSHL	TEMP_CHAR		0749	
		00000000G	EF		01	FB	0049B		CALLS	#1, ENDCHR			
		E0	52		53	F3	004A2	35\$:	AOBLEQ	R3, I, 34\$		0743	
					7E	7C	004A6		CLRQ	-(SP)		0754	
					7E	D4	004A8		CLRL	-(SP)			
		00000000G	EF		03	FB	004AA		CALLS	#3, ENDWRD			
					54	D4	004B1		CLRL	R4		0758	
					64	AC	D5	004B3		TSTL	LINES_BETWEEN		
					02	15	004B6		BLEQ	36\$			
					54	D6	004B8		INCL	R4			
			54	60	AC	C8	004BA	36\$:	BISL2	BREAK_BEFORE_CAPTION, R4			
			42		54	E8	004BE		BLBS	R4, 40\$			
56	4C	A9	02		00	EF	004C1		EXTZV	#0, #2, SCA+196, SCA_HOLD_C_BLDUN		0767	
55	50	A9	02		00	EF	004C7		EXTZV	#0, #2, SCA+200, SCA_HOLD_AC_BLDUN		0768	
			52	20	AC	D0	004CD		MOVL	COUNTER_SPACES_AFTER, R2		0770	
			8F		52	D1	004D1		C MPL	R2, #75			
					04	15	004D8		BLEQ	37\$			
			52	48	8F	9A	004DA		MOVZBL	#75, R2			
					53	D4	004DE	37\$:	CLRL	I			
					11	11	004E0		BRB	39\$			
		4C	A9		03	8A	004E2	38\$:	BICB2	#3, SCA+196		0772	
		50	A9		03	8A	004E6		BICB2	#3, SCA+200		0773	
					20	DD	004EA		PUSHL	#32		0774	
		00000000G	EF		01	FB	004EC		CALLS	#1, ENDCHR			
			53		52	F3	004F3	39\$:	AOBLEQ	R2, I, 38\$		0770	
			02		56	F0	004F7		INSV	SCA_HOLD_C_BLDUN, #0, #2, SCA+196		0778	
4C	A9		02		55	F0	004FD		INSV	SCA_HOLD_AC_BLDUN, #0, #2, SCA+200		0779	
50	A9				7E	7C	00503	40\$:	CLRQ	-(SP)		0783	
					7E	D4	00505		CLRL	-(SP)			
		00000000G	EF		03	FB	00507		CALLS	#3, ENDWRD			
			04	40	AC	E8	0050E		BLBS	CAPTION_IS_CENTERED, 41\$		0787	
			30	44	AC	E9	00512		BLBC	CAPTION_IS_FLUSH_RIGHT, 46\$			
			50		6B	D0	00516	41\$:	MOVL	TSF, R0		0796	
			05	60	AC	E8	00519		BLBS	BREAK_BEFORE_CAPTION, 42\$			
				64	AC	D5	0051D		TSTL	LINES_BETWEEN		0791	
					1E	15	00520		BLEQ	45\$			
			51		6B	D0	00522	42\$:	MOVL	TSF, R1		0793	
			0B	40	AC	E9	00525		BLBC	CAPTION_IS_CENTERED, 43\$		0796	
		50	00	89	04	A0	C3	00529	SUBL3	4(R0), @SCA+120, R0			
			50		02	C6	0052F		DIVL2	#2, R0			
					06	11	00532		BRB	44\$			
		50	00	89	04	A0	C3	00534	43\$:	SUBL3	4(R0), @SCA+120, R0		0798

28	A1		50	D0	0053A	44\$:	MOVL	R0, 40(R1)	0795		
			17	11	0053E		BRB	49\$	0791		
	57	04	A0	D0	00540	45\$:	MOVL	4(R0), HOLD_LM	0804		
			06	11	00544		BRB	47\$	0805		
	0E		54	E8	00546	46\$:	BLBS	R4, 49\$	0816		
	50		6B	D0	00549		MOVL	TSF, R0	0820		
	FC	B9	04	A0	D0	0054C	47\$:	MOVL	4(R0), @SCA+116		
			04	11	00551		BRB	49\$	0733		
	1C	A9	01	D0	00553	48\$:	MOVL	#1, SCA+148	0828		
00000000G		EF	01	D0	00557	49\$:	MOVL	#1, GCA+224	0834		
		05	60	AC	E8	0055E	BLBS	BREAK_BEFORE CAPTION, 50\$	0836		
			64	AC	D5	00562	TSTL	LINES_BETWEEN			
			11	15	00565		BLEQ	51\$			
00000000G		EF	00	FB	00567	50\$:	CALLS	#0, OUTNJ	0839		
			64	AC	D0	0056E	PUSHL	LINES_BETWEEN	0840		
00000000G		EF	01	FB	00571		CALLS	#1, GCSKIP			
		17	5C	AC	E9	00578	51\$:	BLBC	BRN_OPEN, 52\$	0847	
		7E	34	AC	7D	0057C	MOVQ	CAPTION_MAJOR_TYPE, -(SP)	0852		
		00000000G		EF	DD	00580	PUSHL	FOOMRA	0851		
		00000000G		EF	DD	00586	PUSHL	FOOMRA+12			
00000000G		EF	04	FB	0058C		CALLS	#4, PUTTXT			
			3C	AC	DD	00593	52\$:	PUSHL	CAPTION_CASE	0855	
00000000G		EF	01	FB	00596		CALLS	#1, SETCAS			
		58	A9	01	D0	0059D	MOVQ	#1, SCA+208	0859		
00000000G		EF	00	FB	005A1		CALLS	#0, SCANT	0861		
00000000G		8F	00A0	C9	D1	005A8	CMPQ	SCA+280, #RINTES	0867		
				0D	13	005B1	BEQ	53\$			
				7E	7C	005B3	CLRQ	-(SP)	0869		
				7E	D4	005B5	CLRL	-(SP)			
00000000G		EF	03	FB	005B7		CALLS	#3, ENDWRD			
			30	11	005BE		BRB	56\$			
			00DC	C9	D5	005C0	53\$:	TSTL	SCA+340	0872	
			00D4	C9	D5	005C4	BNEQ	56\$			
				C9	D5	005C6	TSTL	SCA+332	0874		
				24	15	005CA	BLEQ	56\$			
		05	EC	B9	E9	005CC	BLBC	@SCA+100, 54\$	0886		
		50		03	D0	005D0	MOVL	#3, R0			
				02	11	005D3	BRB	55\$			
				50	D4	005D5	54\$:	CLRL	R0		
51		50	00D4	C9	C1	005D7	55\$:	ADDL3	SCA+332, R0, CHARS_TO_DROP	0890	
		50	00000000G	EF	D0	005DD	MOVL	MRA, R0			
		0C	A0	51	C2	005E4	SUBL2	CHARS_TO_DROP, 12(R0)	0891		
		04	A0	51	C2	005E8	SUBL2	CHARS_TO_DROP, 4(R0)	0892		
			00D4	C9	D4	005EC	CLRL	SCA+332			
			51	0080	C9	D0	005F0	56\$:	MOVL	SCA+248, HOLD_WRD_PNTR	0896
			52	00D4	C9	D0	005F5	MOVL	SCA+332, HOLD_LST_SP	0897	
				50	D4	005FA	CLRL	I	0899		
				6E40	D0	005FC	57\$:	MOVL	SCA_HOLD[I], SCAE[I]	0900	
F2		88	A940	8F	F3	00602	AOBLEQ	#95, I, 57\$			
			0000005F	6A	D0	0060A	MOVL	PP_SCA, @SCA+100			
		EC	B9	04	AA	D0	0060E	MOVL	PP_SCA+4, @SCA+104		
		F0	B9	08	AA	D0	00613	MOVL	PP_SCA+8, @SCA+108		
		F4	B9	0C	AA	D0	00618	MOVL	PP_SCA+12, @SCA+112		
		F8	B9	10	AA	D0	0061D	MOVL	PP_SCA+16, @SCA+116		
		FC	B9	14	AA	D0	00622	MOVL	PP_SCA+20, @SCA+120		
		00	B9	18	AA	D0	00627	MOVL	PP_SCA+24, @SCA+124		
		04	B9	1C	AA	D0	0062C	MOVL	PP_SCA+28, @SCA+128		
		08	B9								

	0C	B9	20	AA	D0	00631	MOVL	PP_SCA+32, @SCA+132		
	10	B9	24	AA	D0	00636	MOVL	PP_SCA+36, @SCA+136		
	14	B9	28	AA	D0	0063B	MOVL	PP_SCA+40, @SCA+140		
	18	B9	2C	AA	D0	00640	MOVL	PP_SCA+44, @SCA+144		
	0080	C9		51	D0	00645	MOVL	HOLD_WRD_PNTR, SCA+248	0903	
	00D4	C9		52	D0	0064A	MOVL	HOLD_LST_SP, SCA+332	0904	
		04	40	AC	E8	0064F	BLBS	CAPTION_IS_CENTERED, 58\$	0920	
		2C	44	AC	E9	00653	BLBC	CAPTION_IS_FLUSH_RIGHT, 62\$		
		09	00000000G	EF	E8	00657	58\$:	BLBS	GCA+224, 59\$	0923
		05	60	AC	E8	0065E	BLBS	BREAK_BEFORE_CAPTION, 59\$	0930	
			64	AC	D5	00662	TSTL	LINES_BETWEEN		
				1C	15	00665	BLEQ	62\$		
		50		6B	D0	00667	59\$:	MOVL	TSF, R0	0931
		0B	40	AC	E9	0066A	BLBC	CAPTION_IS_CENTERED, 60\$	0934	
51	00	B9	04	A0	C3	0066E	SUBL3	4(R0), @SCA+120, R1		
		51		02	C6	00674	DIVL2	#2, R1		
				06	11	00677	BRB	61\$		
51	00	B9	04	A0	C3	00679	60\$:	SUBL3	4(R0), @SCA+120, R1	0935
	28	A0		51	D0	0067F	61\$:	MOVL	R1, 40(R0)	0933
	00000000G	EF		00	FB	00683	62\$:	CALLS	#0, OUTNJ	0938
			08	AC	DD	0068A	PUSHL	LINES_AFTER	0940	
	00000000G	EF		01	FB	0068D	CALLS	#1, GCSKIP		
	3C	A9		01	D0	00694	MOVL	#1, SCA+180	0942	
			00000000G	FF	DD	00698	PUSHL	@GCA+128	0945	
	00000000G	EF		01	FB	0069E	CALLS	#1, SETCAS		
			00000000G	EF	D4	006A5	CLRL	GCA+60	0947	
	00000000G	FF		58	D0	006AB	MOVL	HOLD_HEADERS, @HCT+8	0953	
				04	006B2	RET			0970	

; Routine Size: 1715 bytes, Routine Base: \$CODE\$ + 0000

```

: 844      0971  1
: 845      0972  1 END      !End of module
: 846      0973  0 ELUDOM

```

PSECT SUMMARY

Name	Bytes	Attributes
\$OWNS	48	NOVEC, WRT, RD, NOEXE, NOSHR, LCL, REL, CON, NOPIC, ALIGN(2)
\$CODE\$	1715	NOVEC, NOWRT, RD, EXE, NOSHR, LCL, REL, CON, NOPIC, ALIGN(2)

Library Statistics

File	Total	Symbols Loaded	Percent	Pages Mapped	Processing Time
_\$255\$DUA28:[SYSLIB]XPORT.L32;1	590	0	0	252	00:00.1

NEWSPAG LIS	NODOPX LIS	OFT LIS	OUTXT LIS
NDXURS LIS	NOTE LIS	OUTLIN LIS	PACK LIS
NM LIS	OUTXHR LIS	NDXXTN LIS	OUTCHA LIS
OUTHDR LIS			