



```
RRRRRRRR    IIIIII    NN      NN    IIIIII    TTTT TTTT TTT
RRRRRRRR    IIIIII    NN      NN    IIIIII    TTTT TTTT TTT
RR          RR      II      II      II      TT
RR          RR      II      II      II      TT
RR          RR      II      II      II      TT
RR          RR      II      II      II      TT
RRRRRRRR    III      NN      NN      NN      TT
RRRRRRRR    III      NN      NN      NN      TT
RR          RR      II      II      NN      TT
RR          RR      II      II      NN      TT
RR          RR      II      II      NN      TT
RR          RR      II      II      NN      TT
RR          RR      IIIIII   NN      NN      IIIIII   TT
RR          RR      IIIIII   NN      NN      IIIIII   TT
```

....  
....  
....  
....

```
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
LL          II      SS
LL          IIIIII   SSSSSSSS
LLLLLLLLLLL IIIIII   SSSSSSSS
```

```

1 0001 0 %TITLE 'RUNOFF initialization'
2 0002 0 MODULE RINIT ( 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: Initialization of RUNOFF.
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   048   REM00048   Ray Marshall   17-May-1984
46 0045 1   Added initialization of GCA_CMD_OSQ to support the insertion
47 0046 1   of output line numbers in the output text.
48 0047 1
49 0048 1   047   REM00047   Ray Marshall   17-November-1983
50 0049 1   Modified the external definition of ATABLE to use the new
51 0050 1   macro ATABLE_DEFINITION defined in ATCODE.REQ.
52 0051 1   Also, changed the call to NATE to use the new literal that
53 0052 1   defines ATABLE's length.
54 0053 1
55 0054 1   046   KFA00046   Ken Alden     10-Oct-1983
56 0055 1   Added init for GCA_CMD_PAGES.
57 0056 1
58 0057 1   045   KFA00045   Ken Alden     15-Jul-1983
59 0058 1   Added init. for GCA_DO_XXX_ALREADY.
60 0059 1
61 0060 1   044   KFA00044   Ken Alden     5-Jul-1983
62 0061 1   Made the sca initialization use literals rather than
63 0062 1   'hard-coded' numbers. This will prevent modifying this
64 0063 1   module when any save/restore expansion is needed.
65 0064 1
66 0065 1   043   KFA00043   Ken Alden     27-Jun-1983
67 0066 1   Added init. for gca_cref_errcnt.
68 0067 1
69 0068 1   042   KAD00042   Keith Dawson  10-Jun-1983
70 0069 1   Fix .NOTE margins to be more sensible. This is motivated by
71 0070 1   the fact that notes come out looking centered on .PS, instead
72 0071 1   of on .LM-.RM as desired.
73 0072 1
74 0073 1   041   KAD00041   Keith Dawson  9-Jun-1983
75 0074 1   For both DSR and DSRPLUS, made default right margin and
76 0075 1   page size (width) 70 instead of 60.
77 0076 1
78 0077 1   040   KAD00040   Keith Dawson  9-May-1983
79 0078 1   Remove support for .DX, .PX.
80 0079 1
81 0080 1   039   KAD00039   Keith Dawson  11-April-1983
82 0081 1   Added support for new termination error messages for
83 0082 1   information written to .BRN file.
84 0083 1
85 0084 1   038   RER00038   Ron Randall  05-Apr-1983
86 0085 1   For DSRPLUS: Improved footnote and topnote initialization.
87 0086 1
88 0087 1   037   KFA00037   Ken Alden     28-Mar-1983
89 0088 1   Altered te_ page initialization to be more logical.
90 0089 1   and easier to follow. now: 6==>entities, 7==>HL's,
91 0090 1   and 17==>Chapter/appendix.
92 0091 1
93 0092 1   036   RER00036   Ron Randall  20-Mar-1983
94 0093 1   For DSRPLUS:
95 0094 1   Added code to initialize topnotes.
96 0095 1   No more passthrough flag...
97 0096 1
98 0097 1   035   KAD00035   Keith Dawson  20-Mar-1983
```

RINIT  
V04-000

RUNOFF initialization  
Revision History

K 15  
16-Sep-1984 01:32:39  
14-Sep-1984 13:07:51

VAX-11 BlISS-32 V4.0-742  
DISK\$VMSMASTER:[RUNOFF.SRC]RINIT.BLI;1 Page 3 (2)

:	99	0098	1	:			
:	100	0099	1	:			Removed LN01 conditionals and all references to .BIX
:	101	0100	1	:			and .BTC files.
:	102	0101	1	:	034	KFA00034	Ken Alden 15-Mar-1983
:	103	0102	1	:			For DSRPLUS: added initialization for passthrough 'lag.
:	104	0103	1	:			For DSR: Enabled save/restore functionality.
:	105	0104	1	:			
:	106	0105	1	:	033	KAD00033	Keith Dawson 07-Mar-1983
:	107	0106	1	:			Global edit of all modules. Updated module names, idents,
:	108	0107	1	:			copyright dates. Changed require files to BLISS library.
:	109	0108	1	:			
:	110	0109	1	:			

```

112 0110 1 %SBTTL 'Module Level Declarations'
113 0111 1
114 0112 1 : TABLE OF CONTENTS:
115 0113 1
116 0114 1 FORWARD ROUTINE
117 0115 1 RINIT : NOVALUE, ! Main initialization routine.
118 0116 1 ECCINI : NOVALUE; ! Initialize ECC structure.
119 0117 1
120 0118 1 : INCLUDE FILES:
121 0119 1
122 0120 1 LIBRARY 'NXPOR:XPORT'; ! XPORT Library
123 0121 1 REQUIRE 'REQ:RNODEF'; ! RUNOFF variant definitions
124 U 0252 1 %IF DSRPLUS %THEN
125 U 0253 1 LIBRARY 'REQ:DPLLIB'; ! DSRPLUS BLISS Library
126 0254 1 %ELSE
127 0255 1 LIBRARY 'REQ:DSRLIB'; ! DSR BLISS Library
128 0256 1 %FI
129 0257 1
130 0258 1
131 0259 1 : EQUATED SYMBOLS:
132 0260 1
133 0261 1 EXTERNAL LITERAL
134 0262 1 RINTES : UNSIGNED (8);
135 0263 1
136 0264 1 : OWN STORAGE:
137 0265 1
138 0266 1 EXTERNAL
139 0267 1 MRAA : FIXED_STRING,
140 0268 1 LSTSF : COUNTED_LIST,
141 0269 1 PDTSF : VECTOR [PDT_SIZE],
142 0270 1 TSFA : VECTOR [TSF_SIZE];
143 0271 1
144 U 0272 1 : GLOBAL STORAGE:
145 U 0273 1
146 0274 1 EXTERNAL
147 0275 1 FRA : FIXED_STRING,
148 0276 1 IRA : FIXED_STRING,
149 0277 1 XMRA : FIXED_STRING,
150 0278 1 MRA, ! Refers to MRAa or MRAb.
151 0279 1 ! Intermediate storage for subtitle and title.
152 0280 1 SBTMRA : FIXED_STRING,
153 0281 1 TITMRA : FIXED_STRING,
154 0282 1 ! Descriptors for subtitle and title.
155 0283 1 SBTTSF : VECTOR [TSF_SIZE],
156 0284 1 TITTSF : VECTOR [TSF_SIZE],
157 0285 1 XTSF : VECTOR [TSF_SIZE], ! Used for <INDEX flag> processing.
158 0286 1 KHAR,
159 0287 1 atable : atable_definition, ! Action table. Used to identify
160 0288 1 ! what type of action is to be
161 0289 1 ! taken on encountering any
162 0290 1 ! given character.
163 0291 1 TSF : REF VECTOR [TSF_SIZE], ! Refers to TSFa or TSFb
164 0292 1 D%ITIM : VECTOR[6], ! Day and time pieces.
165 0293 1 FNESIZ : FN_EXT_SIZE_DEFINITION, ! Sizes of pending footnotes.
166 0294 1 FNISIZ : FN_INT_SIZE_DEFINITION, ! Number of TSF/MRA pairs in each pending footnote.
167 0295 1 FNCT : FNCT_DEFINITION, ! Footnote control table.
168 0296 1 FOOREC : FOOREC_DEFINITION, ! Control area for pass through records

```

```

: 169 0297 1 GCA : GCA_DEFINITION, : Global control area
: 170 0298 1 HCT : HCT_DEFINITION, : Header control area
: 171 0299 1 IFSTK : IFSTACK, : Stack for .IFs, etc
: 172 0300 1 IRAC : IRAC_DEFINITION,
: 173 0301 1 NPAGEN : PAGE_DEFINITION, : Page number of next page
: 174 0302 1 PAGEN : PAGE_DEFINITION, : Page number of current page
: 175 0303 1 BRNOOB : $XPO_IOB (), : IOB for binary output file.
: 176 0304 1 : NDXPOL, : Address of indexing pool, if any.
: 177 0305 1 : NDXSGE, : End of current index segment.
: 178 0306 1 : NDXSGF, : Free space in current index segment.
: 179 0307 1 : XTNCNT, : Number of XTNTAB entries
: 180 0308 1 : XTNLSP, : REF PAGE_DEFINITION, : Page number of last index entry.
: 181 0309 1 : XTNLXS, : REF VECTOR, : Index segment list.
: 182 0310 1 : XTNPOL, : REF BLOCK, : Indexing storage pool.
: 183 0311 1 : XTNSGP, : REF BLOCK, : Address of current segment.
: 184 0312 1 : XTNTAB, : REF XTNTAB_DEFINE, : List of transaction numbers assigned
: 185 0313 1 : XPAGEN, : REF XPAGEN_DEFINE, : Page number associated with transaction number.
: 186 0314 1 FLGT : FLAG_TABLE[FLAG_COUNT],
: 187 0315 1 SCA : SCA_DEFINITION, : Scanner control area.
: 188 0316 1 SCALIT : VECTOR [SCA_CASE_SIZE], : Save case rules here during literal processing.
: 189 0317 1 FS01 : FIXED_STRING, : General purpose character string.
: 190 0318 1 OUTOPT : OUTOPT_DEFINE, : Output options
: 191 0319 1 ECC : $ECC_BLOCKVECTOR,
: 192 0320 1 FRMSTD, : : Depth of FRMSTK (0 means empty)
: 193 0321 1 FRMSTK, : FORM_STACK,
: 194 0322 1 HLC : HLC_DEFINITION,
: 195 0323 1 HLLIST : COUNTED_LIST [MAX_LEVELS],
: 196 0324 1 HLDSP : VECTOR [MAX_LEVELS],
: 197 0325 1 SAVSTK : SAVSTACK,
: 198 0326 1 LSTCHR : REF VECTOR,
: 199 0327 1 LSTLCH : REF VECTOR,
: 200 0328 1 LSTRCH : REF VECTOR,
: 201 0329 1 LSTLDD : REF VECTOR,
: 202 0330 1 LSTCNT : REF COUNTED_LIST,
: 203 0331 1 LSTSKP : REF VECTOR,
: 204 0332 1 CHLST : VECTOR,
: 205 0333 1 LCHLST : VECTOR,
: 206 0334 1 RCHLST : VECTOR,
: 207 0335 1 LDDLST : VECTOR,
: 208 0336 1 NMLST : NUMBER_LIST,
: 209 0337 1 PDT : REF PDT_DEFINITION,
: 210 0338 1 PHAN : PHAN_DEFINITION,
: 211 0339 1 SKPLST : VECTOR [NMLST_MAXSIZE],
: 212 0340 1 TTABLE : COUNTED_LIST [MAX_TABS], : Tab settings
: 213 0341 1 IOBSTK : BLOCKVECTOR [MAX_REQUIRE, IOBSK_LENGTH],
: 214 0342 1 ! Active input file IOBs saved in IOBSTK.
: 215 0343 1 RNEIOB : REF $XPO_IOB (), : Storage for primary input IOB.
: 216 0344 1 RNAIOB : REF $XPO_IOB (), : Storage for output IOB.
: 217 0345 1 RNIIOB : REF $XPO_IOB (), : Input IOB currently active
: 218 0346 1 RNOIOB : REF $XPO_IOB (), : Output file IOB (i.e., for generated document)
: 219 0347 1 FOOIOB : REF $XPO_IOB (), : Standard footnote IOB.
: 220 0348 1 FOTIOB : REF $XPO_IOB (), : Refers to a temporary IOB for footnotes.
: 221 0349 1 FO1IOB : $XPO_IOB (), : Never referred to directly
: 222 0350 1 FO2IOB : $XPO_IOB (), : Never referred to directly
: 223 0351 1 TTEIOB : $XPO_IOB (), : Standard error file (usually TT:)
: 224 0352 1 TTIIOB : $XPO_IOB (), : Standard input file (usually TTY:)
: 225 0353 1 TTOIOB : $XPO_IOB (), : Standard output file (usually TTY:)

```





```

: 257 0384 1 %SBTTL 'RINIT -- body of main routine'
: 258 0385 1 GLOBAL ROUTINE rinit : NOVALUE =
: 259 0386 1
: 260 0387 1 !++
: 261 0388 1 ! FUNCTIONAL DESCRIPTION:      RUNOFF initialization
: 262 0389 1 !
: 263 0390 1 ! FORMAL PARAMETERS:      None
: 264 0391 1 !
: 265 0392 1 ! IMPLICIT INPUTS:      None
: 266 0393 1 !
: 267 0394 1 ! IMPLICIT OUTPUTS:     None
: 268 0395 1 !
: 269 0396 1 ! ROUTINE VALUE:
: 270 0397 1 ! COMPLETION CODES:     None
: 271 0398 1 !
: 272 0399 1 ! SIDE EFFECTS:        None
: 273 0400 1 ! --
: 274 0401 1
: 275 0402 2     BEGIN
: 276 0403 2     LOCAL
: 277 0404 2         status;
: 278 0405 2
: 279 0406 2     !
: 280 0407 2     ! Initialize fixed_strings.
: 281 0408 2
: 282 0409 2     fs_init (ira);
: 283 0410 2     fs_init (mraa);
: 284 0411 2     fs_init (fra);
: 285 0412 2     fs_init (fs01);
: 286 0413 2     fs_init (titmra);
: 287 0414 2     fs_init (sbtmra);
: 288 0415 2     fs_init (xmra);
: 289 0416 2
: 290 0417 2     !
: 291 0418 2     ! Initialize text descriptors.
: 292 0419 2
: 293 0420 2     INCR I FROM 0 TO TSF_SIZE - 1 DO
: 294 0421 3         BEGIN
: 295 0422 3             TSFA [I] = 0;           ! Text descriptor for primary area
: 296 0423 3             SBTTSF [I] = 0;       ! Text descriptor for subtitles
: 297 0424 3             TITTSF [I] = 0;      ! Text descriptor for titles
: 298 0425 3             XTSTF [I] = 0;       ! Text descriptor for indexing
: 299 0426 3         END;
: 300 0427 2
: 301 0428 2 ! ***Begin SAVE/RESTORE reallocation
: 302 0429 2     SAVINI ();
: 303 0430 2 ! ***End SAVE/RESTORE
: 304 0431 2
: 305 0432 2     !
: 306 0433 2     ! Initialize draft flags in all TSF's to space (%C' ').
: 307 0434 2
: 308 0435 2     TSF = TSFA;
: 309 0436 2     TSF_DRAFT_FLAG = %C' ';
: 310 0437 2     TSF = SBTTSF;
: 311 0438 2     TSF_DRAFT_FLAG = %C' ';
: 312 0439 2     TSF = TITTSF;
: 313 0440 2     TSF_DRAFT_FLAG = %C' ';

```

```

314      0441      2      TSF = XTSF;
315      0442      2      TSF_DRAFT_FLAG = %C' ';
316      0443      2      |
317      0444      2      | Point to primary text areas.
318      0445      2      |
319      0446      2      MRA = MRAA;           ! Primary intermediate build area.
320      0447      2      TSF = TSFA;         ! Primary control area.
321      0448      2      |
322      0449      2      NMLST_COUNT = 0;     ! No waiting number lists
323      0450      2      GCA_FEHLER = FALSE; ! No errors yet
324      0451      2      GCA_ERRCNT = 0;     ! No errors yet
325      0452      2      GCA_ERR_DIR = REPORT_ERR_STD; ! Report errors in error log.
326      0453      2      GCA_CMD_MSG = REPORT_ERR_BOTH; ! After opening files report errors in both places.
327      0454      2      |
328      0455      2      | Some I/O initialization.
329      0456      2      |
330      0457      2      RNIOB = .RNEIOB;     ! Set up pointer to primary input IOB.
331      0458      2      RNOIOB = .RNAIOB;    ! Set up pointer to output IOB.
332      0459      2      $XPO_IOB_INIT (IOB = TTIIOB); ! Standard input (TTY:)
333      0460      2      $XPO_IOB_INIT (IOB = TTOIOB); ! Standard output (TTY:)
334      0461      2      $XPO_IOB_INIT (IOB = TTEIOB); ! Standard error log (TTY:)
335      0462      2      $XPO_IOB_INIT (IOB = TSIIOB); ! Standard input, but STREAM.
336      0463      2      $XPO_IOB_INIT (IOB = BRNOOB); ! Binary-output IOB.
337      P 0464      2      $XPO_IOB_INIT (IOB = F01IOB, ! Standard footnote IOB.
338      P 0465      2      FILE_SPEC = '001RNO.TMP',
339      0466      2      ATTRIBUTES = BINARY);
340      P 0467      2      $XPO_IOB_INIT (IOB = F02IOB, ! Temporary working footnote IOB.
341      P 0468      2      FILE_SPEC = '002RNO.TMP',
342      0469      2      ATTRIBUTES = BINARY);
343      0470      2      F00IOB = F01IOB;     ! Set up "double IOBing"
344      0471      2      F01IOB = F02IOB;
345      0472      2      |
346      U 0473      2      %IF DSRPLUS %THEN
347      U 0474      2      |
348      U 0475      2      | Initialize topnote control tables and file access tables.
349      U 0476      2      |
350      U 0477      2      INCR I FROM 0 TO (TN_CONTROL - 1) DO
351      U 0478      2      TOPNOT [.I] = 0;
352      U 0479      2      |
353      U 0480      2      INCR I FROM 0 TO (TNREC_SIZE - 1) DO
354      U 0481      2      TNREC [.I] = 0;
355      U 0482      2      |
356      U 0483      2      |
357      U 0484      2      | Initialize topnote IOBs.
358      U 0485      2      |
359      U 0486      2      $XPO_IOB_INIT (IOB = TNOIOB,
360      U 0487      2      FILE_SPEC = '001RN0TNO.TMP',
361      U 0488      2      ATTRIBUTES = BINARY);
362      U 0489      2      $XPO_IOB_INIT (IOB = TN1IOB,
363      U 0490      2      FILE_SPEC = '001RN0TN1.TMP',
364      U 0491      2      ATTRIBUTES = BINARY);
365      U 0492      2      |
366      U 0493      2      | Set up "double IOBing".
367      U 0494      2      |
368      U 0495      2      TNUIOB = TNOIOB;
369      U 0496      2      TN1IOB = TN1IOB;
370      U 0497      2      %FI

```

```

: 371      0498      2
: 372      0499      2
: 373      0500      2      | Open standard error log
: 374      0501      2
: 375      P 0502      2      status = $XPO_OPEN ( IOB = TTEIOB
: 376      P 0503      2      ,FILE_SPEC = $XPO_ERROR
: 377      P 0504      2      ,OPTIONS = OUTPUT
: 378      0505      2      );
: 379      0506      2
: 380      0507      2      | Open user's terminal for input
: 381      0508      2
: 382      P 0509      2      status = $XPO_OPEN ( IOB = TTIIOB
: 383      P 0510      2      ,FILE_SPEC = $XPO_INPUT
: 384      P 0511      2      ,OPTIONS = INPUT
: 385      0512      2      );
: 386      0513      2
: 387      0514      2      | Open user's terminal for output
: 388      0515      2
: 389      P 0516      2      status = $XPO_OPEN ( IOB = TTOIOB
: 390      P 0517      2      ,FILE_SPEC = $XPO_OUTPUT
: 391      P 0518      2      ,OPTIONS = OUTPUT
: 392      0519      2      );
: 393      0520      2
: 394      0521      2      | Initialize HCT (header control table)
: 395      0522      2
: 396      0523      2      HCT_HD_CASE = HCT_HD_MIXED;           ! Display 'Page' and 'Index'
: 397      0524      2      HCT_HEADERS = TRUE;
: 398      0525      2      HCT_NUMBER_PAGE = TRUE;
: 399      0526      2      HCT_NMPG_NP = TRUE;
: 400      0527      2      HCT_SUBTITLE = FALSE;
: 401      0528      2
: 402      U 0529      2      %IF DSRPLUS %THEN
: 403      U 0530      2      HCT_TITLE_ALWAYS = FALSE;           ! FIRST TITLE [ALWAYS]
: 404      0531      2      %FI
: 405      0532      2
: 406      0533      2      HCT_LAYOUT = LAYOUT_STANDARD;           ! Standard page arrangement.
: 407      0534      2      HCT_LAYOUTN = 0;
: 408      0535      2      HCT_LAYOUT_NP = LAYOUT_STANDARD;
: 409      0536      2      HCT_LAYOUTN_NP = 0;
: 410      0537      2      HCT_ODD_EVEN = 0;                       ! Set to even parity, as each call on NEWPAG
: 411      0538      2      ! flips the parity before outputting the title.
: 412      0539      2
: 413      0540      2      | Initialize footnote control tables.
: 414      0541      2
: 415      0542      2      INCR I FROM 0 TO FNCT_CONTROL - 1 DO
: 416      0543      2      FNCT [I] = 0;
: 417      0544      2
: 418      0545      2      INCR I FROM 0 TO FNCT_MAX - 1 DO
: 419      0546      2      BEGIN
: 420      0547      2      FNESIZ [I] = 0;
: 421      0548      2      FNISIZ [I] = 0;
: 422      0549      2      END;
: 423      0550      2
: 424      0551      2
: 425      0552      2      | Initialize 'passthrough' work area.
: 426      0553      2
: 427      0554      2      INCR I FROM 0 TO FOOREC_SIZE - 1 DO
```

```

428 0555 2          FOOREC [.I] = 0;
429 0556 2
430 0557 2
431 0558 2          Initialize tables for variables (.VARIABLE and /VARIANT)
432 0559 2
433 0560 2          VRCNT = 0;                                ! There are no variables defined yet.
434 0561 2          VRTFLG [0] = %C' ';
435 0562 2          VRFFLG [0] = %C' ';
436 0563 2          VRBOOL [0] = TRUE;
437 0564 2
438 0565 2          Initialize the IFSTACK
439 0566 2
440 0567 2          IFSTK [0, IFSTK_DEPTH] = 0;                ! There are no active ".IFs"
441 0568 2          IFSTK [0, IFSTK_REQ_D] = 0;
442 0569 2          IFSTK [0, IFSTK_BOO] = TRUE;
443 0570 2          IFSTK [0, IFSTK_VR] = 0;
444 0571 2          IFSTK [0, IFSTK_SOURCE] = 0;
445 0572 2          IFSTK [0, IFSTK_FLAG] = %C' ';
446 0573 2
447 0574 2          Initialize .LIST ELEMENT character "counter".
448 0575 2
449 0576 2          LSTCHR = CHLST;
450 0577 2          LSTCHR [0] = 0;
451 0578 2
452 0579 2          Initialize list of .LIST ELEMENT counters
453 0580 2
454 0581 2          LSTCNT = LSTSTF;
455 0582 2
456 0583 2          Initialize .LIST ELEMENT display information
457 0584 2
458 0585 2          LSTLCH = LCHLST;
459 0586 2          LSTRCH = RCHLST;
460 0587 2          LSTLDD = LDDLST;
461 0588 2          LSTLCH [0] = 0;
462 0589 2          LSTRCH [0] = %C' ';
463 0590 2          LSTLDD [0] = TCONVRT_DEC_NOZ;
464 0591 2          LSTCNT [CL_MAX_INDEX] = CL_DEFAULT_SIZE;
465 0592 2          LSTCNT [CL_INDEX] = 1;
466 0593 2          LSTCNT [1] = 0;
467 0594 2
468 0595 2          Initialize inter-list-element spacing;
469 0596 2
470 0597 2          LSTSKP = SKPLST;
471 0598 2          LSTSKP [0] = 1;
472 0599 2
473 0600 2          Initialize list of .HEADER LEVEL counters
474 0601 2
475 0602 2          HLLIST [CL_MAX_INDEX] = 6;
476 0603 2          HLLIST [CL_INDEX] = 1;
477 0604 2
478 0605 2
479 0606 2          Set display characteristics for all header level counters to decimal
480 0607 2
481 0608 2          INCR I FROM 1 TO .HLLIST [CL_MAX_INDEX] DO
482 0609 2              BEGIN
483 0610 2                  HLDSP [I - 1] = TCONVRT_DEC_NOZ;
484 0611 2                  HLLIST [.I] = 0;

```

```

485 0612 2      END;
486 0613 2
487 0614 2
488 0615 2      Initialize list of .SAVE/RESTORE variables.      ! ***SAVE/RESTORE
489 0616 2
490 0617 2      Initialize the SAVSTACK
491 0618 2
492 0619 2      SAVSTK [0, SAVSTK_DEPTH] = 0;      ! There are no active ".SAVES"
493 0620 2      SAVSTK [0, SAVSTK_REQ D] = 0;
494 0621 2      SAVSTK [0, SAVSTK_ADDRESS] = 0;
495 0622 2      SAVSTK [0, SAVSTK_SOURCE] = 0;
496 0623 2
497 0624 2
498 0625 2      Initialize the entire ECC structure. The TRUE third argument causes
499 0626 2      us to zero the entity counters as well.
500 0627 2
501 0628 2      ECCINI (EXAMP_OFFSET, APPEND_OFFSET, TRUE);
502 0629 2
503 0630 2      Initialize .HEADER LEVEL processing controls.
504 0631 2
505 0632 2      HLC_CO_NBITS = FALSE;      ! Don't carry over underlining and bolding to subtitles,
506 0633 2      HLC_CO_OVR = TRUE;      ! but do carry over overstriking.
507 0634 2      HLC_UPPER = 1;      ! Force .HLs to upper case.
508 0635 2      HLC_CAP = MAX_LEVELS;      ! First character of all .HLs to upper case.
509 0636 2      HLC_RUNON = 3;      ! Generate '-' starting with .HL3
510 0637 2      HLC_HEADLC = 1;      ! Skip this many lines after the .HL command
511 0638 2      HLC_HEADLT = 7;      ! Anticipate this many lines when doing a test page
512 0639 2      HLC_HEADLB = 3;      ! Skip this many lines before the .HL command
513 0640 2      HLC_HEADSP = 2;      ! Insert this many spaces after the section number.
514 0641 2      HLC_NO NUMBER = 99;      ! All headers are numbered.
515 0642 2      HLC_CENTER = 99;      ! Don't center any headers.
516 0643 2
517 0644 2      Initialize tab settings;
518 0645 2
519 0646 2      TTABLE [CL_MAX_INDEX] = MAX_TABS;
520 0647 2      TTABLE [1] = 8;
521 0648 2
522 0649 2      INCR I FROM 2 TO MAX_TABS DO
523 0650 2          TTABLE [.I] = 8 * TTABLE [.I - 1];
524 0651 2
525 0652 2      TTABLE [CL_INDEX] = MAX_TABS;
526 0653 2
527 0654 2      End of initialization of tab stops
528 0655 2
529 0656 2
530 0657 2
531 0658 2      Normal character recognition
532 0659 2
533 0660 2      INCR I FROM 0 TO (atable_length - 1) DO
534 0661 2          NATE (.I);
535 0662 2
536 0663 2
537 0664 2      Set up case translation rules ("as is in input", i.e., .UPPER CASE)
538 0665 2
539 0666 2      SETCAS (LEAVE_CASE);
540 0667 2
541 0668 2      Initialize flag table. Set initial enabled or disabled states.

```

```

: 542 0669 2 !
: 543 0670 22 FLGT [SUB_FLAG, FLAG_ENABLED] = FALSE; ! SUBSTITUTE flag
: 544 0671 22 FLGT [EFO_FLAG, FLAG_ENABLED] = TRUE; ! ENDFOOTNOTE flag
: 545 0672 22 FLGT [CON_FLAG, FLAG_ENABLED] = TRUE; ! CONTROL flag
: 546 0673 22 FLGT [QUO_FLAG, FLAG_ENABLED] = TRUE; ! QUOTE flag
: 547 0674 22 FLGT [UPP_FLAG, FLAG_ENABLED] = TRUE; ! UPPERCASE flag
: 548 0675 22 FLGT [LOW_FLAG, FLAG_ENABLED] = TRUE; ! LOWERCASE flag
: 549 0676 22 FLGT [CAP_FLAG, FLAG_ENABLED] = FALSE; ! CAPITALIZE flag
: 550 0677 22 FLGT [UND_FLAG, FLAG_ENABLED] = TRUE; ! UNDERLINE flag
: 551 0678 22 FLGT [BLD_FLAG, FLAG_ENABLED] = FALSE; ! BOLD flag
: 552 0679 22 FLGT [SPA_FLAG, FLAG_ENABLED] = TRUE; ! SPACE flag
: 553 0680 22 FLGT [IND_FLAG, FLAG_ENABLED] = FALSE; ! INDEX flag
: 554 0681 22 FLGT [HYP_FLAG, FLAG_ENABLED] = FALSE; ! HYPHENATE flag
: 555 0682 22 FLGT [OVR_FLAG, FLAG_ENABLED] = FALSE; ! OVERSTRIKE flag
: 556 0683 22 FLGT [SBX_FLAG, FLAG_ENABLED] = TRUE; ! SUBINDEX flag
: 557 0684 22 FLGT [COM_FLAG, FLAG_ENABLED] = TRUE; ! COMMENT flag
: 558 0685 22 FLGT [PER_FLAG, FLAG_ENABLED] = FALSE; ! PERIOD flag
: 559 0686 22 FLGT [BRK_FLAG, FLAG_ENABLED] = FALSE; ! BREAK flag
: 560 0687 22
: 561 U 0688 22 %IF DSRPLUS %THEN
: 562 U 0689 22 FLGT [NPX_FLAG, FLAG_ENABLED] = FALSE; ! No-permute-index flag
: 563 0690 22 %FI
: 564 0691 22
: 565 0692 22
: 566 0693 22 ! Set up default flag characters for all flags.
: 567 0694 22
: 568 0695 22 FLGT [SUB_FLAG, FLAG_CHARACTER] = %C'$'; ! SUBSTITUTE flag
: 569 0696 22 FLGT [EFO_FLAG, FLAG_CHARACTER] = %C'!'; ! ENDFOOTNOTE flag
: 570 0697 22 FLGT [CON_FLAG, FLAG_CHARACTER] = %C'.'; ! CONTROL flag
: 571 0698 22 FLGT [QUO_FLAG, FLAG_CHARACTER] = %C' '; ! QUOTE flag
: 572 0699 22 FLGT [UPP_FLAG, FLAG_CHARACTER] = %C'*'; ! UPPERCASE flag
: 573 0700 22 FLGT [LOW_FLAG, FLAG_CHARACTER] = %C'\'; ! LOWERCASE flag
: 574 0701 22 FLGT [CAP_FLAG, FLAG_CHARACTER] = %C'<'; ! CAPITALIZE flag
: 575 0702 22 FLGT [UND_FLAG, FLAG_CHARACTER] = %C'&'; ! UNDERLINE flag
: 576 0703 22 FLGT [BLD_FLAG, FLAG_CHARACTER] = %C'*'; ! BOLD flag
: 577 0704 22 FLGT [SPA_FLAG, FLAG_CHARACTER] = %C'#'; ! SPACE flag
: 578 0705 22 FLGT [IND_FLAG, FLAG_CHARACTER] = %C'>'; ! INDEX flag
: 579 0706 22 FLGT [HYP_FLAG, FLAG_CHARACTER] = %C'='; ! HYPHENATE flag
: 580 0707 22 FLGT [OVR_FLAG, FLAG_CHARACTER] = %C'~'; ! OVERSTRIKE flag
: 581 0708 22 FLGT [SBX_FLAG, FLAG_CHARACTER] = %C'>'; ! SUBINDEX flag
: 582 0709 22 FLGT [COM_FLAG, FLAG_CHARACTER] = %C'!'; ! COMMENT flag
: 583 0710 22 FLGT [PER_FLAG, FLAG_CHARACTER] = %C'+'; ! PERIOD flag
: 584 0711 22 FLGT [BRK_FLAG, FLAG_CHARACTER] = %C'!'; ! BREAK flag
: 585 0712 22
: 586 U 0713 22 %IF DSRPLUS %THEN
: 587 U 0714 22 FLGT [NPX_FLAG, FLAG_CHARACTER] = %C' '; ! No-permute-index flag
: 588 0715 22 %FI
: 589 0716 22
: 590 0717 22
: 591 0718 22 ! Update action table to be consistent with the flag table.
: 592 0719 22
: 593 0720 22 ATABLE [.FLGT [QUO_FLAG, FLAG_CHARACTER]] = A_FLAG; ! Enable QUOTE flag
: 594 0721 22 ATABLE [.FLGT [UPP_FLAG, FLAG_CHARACTER]] = A_FLAG; ! Enable UPPERCASE flag
: 595 0722 22 ATABLE [.FLGT [LOW_FLAG, FLAG_CHARACTER]] = A_FLAG; ! Enable LOWERCASE flag
: 596 0723 22 ATABLE [.FLGT [UND_FLAG, FLAG_CHARACTER]] = A_FLAG; ! Enable UNDERLINE flag
: 597 0724 22 ATABLE [.FLGT [SPA_FLAG, FLAG_CHARACTER]] = A_FLAG; ! Enable SPACE flag
: 598 0725 22

```

```

: 599      0726      2      | Initialize IRAC
: 600      0727      2      |
: 601      0728      2      | IRAC_IPAGEN = 1;           ! Input starts on page 1.
: 602      0729      2      | IRAC_ISEQN = 0;
: 603      0730      2      | IRAC_BARS = FALSE;       ! Don't want change bars.
: 604      0731      2      | IRAC_DRAFT_FLG = %C' ';  ! Default /DRAFT flag.
: 605      0732      2      |
: 606      0733      2      | Initialize SCA
: 607      0734      2      |
: 608      0735      2      | BEGIN
: 609      0736      2      | MAP
: 610      0737      2      |   SCA : VECTORE[SCA_SIZE],
: 611      0738      2      |   MRA : REF FIXED_STRING;
: 612      0739      2      |
: 613      0740      2      |
: 614      0741      2      | The following initialization block is different for DSRPLUS because the storage
: 615      0742      2      | area where some SCA words are stored are different. This means that if we say,
: 616      0743      2      | 'SCA[I] = 0 for all the SCA, then the special (SAVED) ones would have a storage
: 617      0744      2      | address of zero.
: 618      0745      2      |
: 619      0746      2      | INCR I FROM 0 TO (SCA_SAVE_START - 1) DO SCA[I] = 0;
: 620      0747      2      | SCA_CC_OK = FALSE;
: 621      0748      2      | SCA_KER = FALSE;
: 622      0749      2      | INCR I FROM (SCA_SAVE_END + 1) TO (SCA_SIZE - 1) DO SCA[I] = 0;
: 623      0750      2      | SCA_FC_CASE = TRUE;
: 624      0751      2      | SCA_RM = 70;              ! Changed from 60 by KAD, 9-Jun-1983.
: 625      0752      2      | SCA_LM = 0;
: 626      0753      2      | SCA_SPACING = 1;
: 627      0754      2      | SCA_FC = TRUE;
: 628      0755      2      | SCA_FILL = TRUE;
: 629      0756      2      | SCA_JUSTIFY = TRUE;
: 630      0757      2      | SCA_CROCK = TRUE;
: 631      0758      2      | SCA_PERIOD = TRUE;
: 632      0759      2      | SCA_FLAGS = TRUE;
: 633      0760      2      | SCA_DO_UND = TRUE;
: 634      0761      2      | SCA_DO_BLD = TRUE;
: 635      0762      2      | SCA_DO_OVR = TRUE;
: 636      0763      2      | SCA_DO_HYP = TRUE;
: 637      0764      2      | SCA_DO_IND = TRUE;
: 638      0765      2      | SCA_AUTOTITLE = TRUE;    ! AUTOTITLE in effect
: 639      0766      2      | SCA_SECT_EMPTY = TRUE;  ! No HEADER LEVELs with text yet.
: 640      0767      2      | SCA_INDEX = TRUE;       ! Obey indexing commands.
: 641      0768      2      | SCA_XROUTINE = FALSE;   ! FALSE ==> use XR as the indexing routine.
: 642      0769      2      | SCA_BAR_CHAR = %C' ';   ! Default change bar character.
: 643      0770      2      | SCA_WRD_PNTR = .FS START (MRA);
: 644      0771      2      | SCA_WRD_CPEND = RINTES;
: 645      0772      2      |
: 646      0773      2      | End of initialization of SCA
: 647      0774      2      |
: 648      0775      2      | END;
: 649      0776      2      |
: 650      0777      2      |
: 651      0778      2      | Initialize OUTOPT
: 652      0779      2      |
: 653      0780      2      | OUTOPT_UND_CHAR = %C' ';
: 654      0781      2      | OUTOPT_UND_NOSP = FALSE; ! Underline character is not non-spacing.
: 655      0782      2      | OUTOPT_UND_SEP = FALSE; ! Don't use a separate line for

```

```

656 0783 2
657 0784 2 OUTOPT_OVER = TRUE;
658 0785 2 OUTOPT_BACK = FALSE;
659 0786 2
660 0787 2 Pick up date and time
661 0788 2
662 0789 2 RNODAT (DATTIM);
663 0790 2
664 0791 2 Initialize GCA
665 0792 2
666 0793 2 GCA_AUTOSUBT = TRUE;
667 0794 2 GCA_AUTOPARA = FALSE;
668 0795 2 GCA_AUTOTABL = FALSE;
669 0796 2 GCA_AUTOJUST = TRUE;
670 0797 2 GCA_KEEP = FALSE;
671 0798 2 GCA_MAX_ASTN = 1;
672 0799 2 GCA_BARS = FALSE;
673 0800 2 GCA_BARS_ENAB = FALSE;
674 0801 2 GCA_BARS_ON = FALSE;
675 0802 2 GCA_EVER_BARS = FALSE;
676 0803 2 GCA_CMD_BAR = TRUE;
677 0804 2 GCA_CMD_RIT = FALSE;
678 0805 2 GCA_CMD_PAGES = FALSE;
679 0806 2 GCA_DEBUG_CND = FALSE;
680 0807 2 GCA_DEBUG_INDEX = FALSE;
681 0808 2 GCA_DEBUG_TOC = FALSE;
682 0809 2 GCA_DEBUG_FIL = FALSE;
683 0810 2 GCA_CMD_UND = TRUE;
684 0811 2 GCA_CMD_BLD = TRUE;
685 0812 2 GCA_CMD_OVR = TRUE;
686 0813 2 GCA_CMD_IND = TRUE;
687 0814 2 GCA_BIX = FALSE;
688 0815 2 GCA_CMD_BTC = FALSE;
689 0816 2 GCA_BTC = FALSE;
690 0817 2 GCA_PWIDTH = 150;
691 0818 2 GCA_LWIDTH = 70;
692 0819 2
693 0820 2 GCA_NOTE_PRIM = 8;
694 0821 2
695 0822 2 GCA_NOTE_ALT = 4;
696 0823 2 GCA_SKIPPING = FALSE;
697 0824 2 GCA_NORMAL_XTN = 1;
698 0825 2 GCA_FOOT_XTN = 1;
699 0826 2 GCA_INDEX_FMT = TRUE;
700 0827 2
701 U 0828 2 %IF DSRPLUS %THEN
702 U 0829 2 GCA_XCASE = LEAVE_CASE;
703 0830 2 %ELSE
704 0831 2 GCA_XCASE = ONE_CAP;
705 0832 2 %FI
706 0833 2
707 0834 2 GCA_MAX_REQUIRE = MAX_REQUIRE;
708 0835 2 GCA_REQ_DEPTH = 0;
709 0836 2 gca_page_cnt = 0;
710 0837 2 gca_contents_cnt = 0;
711 0838 2 gca_index_cnt = 0;
712 0839 2 gca_new_cref_cnt = 0;

```

! underlining.  
! Do special functions by overprinting.  
! Do not use the backspace character.

! Default is .AUTOSUBTITLE 1  
! Default is .NO AUTOPARAGRAPH  
! Default is .NO AUTOTABLE  
! Default is .AUTOJUSTIFY  
! Default is .NO KEEP

! Don't want change bars.

! Change bars never seen.  
! Assume user won't forbid bars.  
! Don't shift /RIGHT.  
! Don't assume /PAGES.  
! Don't assume /DEBUG:CONDITIONALS  
! Don't assume /DEBUG:INDEX  
! Don't assume /DEBUG:CONTENTS  
! Don't assume /DEBUG:FILES  
! Assume user won't forbid underlining.  
! Assume user won't forbid bolding.  
! Assume user won't forbid overstriking.  
! Assume user won't forbid indexing.  
! Assume user won't say /BIX  
! Assume user won't say /BTC

! Changed from 60 by KAD, 9-Jun-1983.

! Primary margins for .NOTE  
! Changed from 15 by KAD, 10-Jun-1983.

! Secondary margins for .NOTE  
! There are no active ".IFs"

! Generate DOTS in the index

! For DSRPLUS, default index case rule is stet-case.

! For DSR, default index case rule is cap. first letter.

! No .REQUIRES yet (i.e., main file)  
! Have no physical pages yet.  
! No table-of-contents records.  
! No index records.  
! No output .BRN records.



```

713 0840 2 gca_old_cref_cnt = 0; ! No input .BRN records.
714 0841 2 !
715 0842 2 gca_black_box = 0; ! Not /AUTOMATIC processing.
716 0843 2 gca_cref_errcnt = 0; ! No unresolved crefs yet.
717 0844 2 gca_cross_reference = 0; ! No cross-referencing.
718 0845 2 gca_expand_cref = 0; ! No input .BRN file for cross-referencing.
719 0846 2 gca_old_brn_exists = 0; !
720 0847 2 gca_cref_changed = 0; ! No changes yet in cross-references.
721 0848 2 gca_do_cont_already = false; ! No .DO CONTENTS commands have been hit.
722 0849 2 gca_do_index_already = false; ! No .DO INDEX commands have been hit.
723 0850 2 !
724 0851 2 GCA_PCHAX = FALSE; ! No pending .NUMBER CHAPTER/APPENDIX.
725 0852 2 GCA_FLAG_CMD = H_FLAGS_ALL;
726 0853 2 GCA_LINE_PEND = 0; ! No lines pending.
727 0854 2 GCA_CASE = LEAVE CASE;
728 0855 2 GCA_CMD_ISQ = FALSE; ! Don't output input line sequence numbers.
729 0856 2 GCA_CMD_OSQ = FALSE; ! Don't output output line sequence numbers.
730 0857 2 GCA_ORANGE_CNT = 0; ! No output pages specified.
731 0858 2 GCA_SKIP_OUT = FALSE; ! Start outputting immediately.
732 0859 2 GCA_CMD_QUICK = FALSE; ! Assume user won't say /QUICK.
733 0860 2
734 U 0861 2 %IF (%BLISS(BLISS36) AND DSRPLUS) %THEN !
735 U 0862 2 GCA_TWIDTH_36 = -1; ! Flag to tell if terminal width is changed.
736 U 0863 2 GCA_TESCAPE_36 = 0; ! Terminal escape characteristics.
737 0864 2 %FI
738 0865 2
739 0866 2 !
740 0867 2 ! Initialize PHAN
741 0868 2 !
742 0869 2 PHAN_PLINES = 66; ! Can have this many lines per page at most,
743 0870 2
744 0871 2 %IF %BLISS (BLISS32) %THEN
745 0872 2 PHAN_SLINES = 66;
746 U 0873 2 %ELSE
747 U 0874 2 PHAN_SLINES = 60; ! but the spooler allows only this many,
748 0875 2 %FI
749 0876 2
750 0877 2 PHAN_LLINES = 58; ! and I'm going to let you try to write this many.
751 0878 2 PHAN_TOP_PAGE = TRUE; ! Start at top of a new page
752 0879 2 PHAN_LINES_TP = 0;
753 0880 2 PHAN_FIGURE = 0; ! No waiting figures
754 0881 2 PHAN_TOP_FIRST = TRUE; ! Output starts at top of first page.
755 0882 2 PHAN_HEADER = FALSE; ! Don't want header at top of first page unless .FIRST TITLE
756 0883 2 PHAN_FORM_PEND = FALSE; ! No pending form feed.
757 0884 2 PHAN_PAGING = TRUE; ! Divide document into pages.
758 0885 2 PHAN_CMD_PAGING = TRUE; ! Assume /PAGING
759 0886 2 PHAN_RIGHT = 0; ! Don't shift output right.
760 0887 2 PHAN_DOWN = 0; ! Don't move output down on page.
761 0888 2 PHAN_SIMULATE = FALSE; ! Don't simulate formfeeds.
762 0889 2 PHAN_BOTTOM = 0; ! Nothing on page yet.
763 0890 2 PHAN_PAUSE = FALSE; ! Don't assume /PAUSE
764 0891 2 PHAN_LCHAR = 0; ! No character to the left of the page number
765 0892 2 PHAN_RCHAR = 0; ! No character to the right of the page number
766 0893 2
767 0894 2 !
768 0895 2 ! Initialize current page number
769 0896 2 !

```

```

770 0897 BEGIN
771 0898 MAP
772 0899     PAGEN : VECTOR [PAGE_SCT_SIZE];
773 0900
774 0901 INCR I FROM 0 TO PAGE_SCT_SIZE - 1 DO
775 0902     PAGEN [I] = 0;
776 0903
777 0904 END;
778 0905
779 0906 PAGEN [SCT_PAGE] = 1;
780 0907 PAGEN [SCT_RUN_PAGE] = 1;
781 0908
782 0909     Set up initial display characteristics for the various parts of the page
783 0910     number.
784 0911
785 0912     PAGEN [SCT_PAGE_D] = TCONVRT_DEC_NOZ;           ! Page number part to decimal, no leading zeros.
786 0913     PAGEN [SCT_INDEX_D] = TCONVRT_LET_MIX;         ! The word index appears as "index"
787 0914     PAGEN [SCT_CHAPT_D] = TCONVRT_DEC_NOZ;         ! Chapters to decimal, no leading zeroes.
788 0915     PAGEN [SCT_APPEN_D] = TCONVRT_LET_UPP;       ! Appendices to upper case letters.
789 0916     PAGEN [SCT_SUBPG_D] = TCONVRT_LET_UPP;       ! Subpages to upper case letters.
790 0917
791 0918
792 0919     Initialize next page number
793 0920
794 0921 BEGIN
795 0922 MAP
796 0923     NPAGEN : VECTOR [PAGE_SCT_SIZE],
797 0924     PAGEN  : VECTOR [PAGE_SCT_SIZE];
798 0925
799 0926 INCR I FROM 0 TO PAGE_SCT_SIZE - 1 DO
800 0927     NPAGEN [I] = .PAGEN [I];
801 0928
802 0929 END;
803 0930
804 0931 NPAGEN [SCT_PAGE] = 2;
805 0932 NPAGEN [SCT_RUN_PAGE] = 2;
806 0933
807 0934     Initialize paragraph description table.
808 0935
809 0936 PDT = PDTSTF;           ! Point to paragraph information.
810 0937 PDT_IP = 2;           ! Make sure this many lines are
811 0938                       ! available on current page,
812 0939 PDT_SKIP = 1;         ! then skip this many lines unless
813 0940                       ! at top of page or section is empty,
814 0941 PDT_INDENT = 5;       ! and finally, indent this much,
815 0942
816 0943
817 0944     Delete leftover index entries, if any.
818 0945
819 0946 IF .NDXPOL NEQ 0
820 0947 THEN
821 0948     BEGIN
822 0949     FPOOL (NDXPOL);
823 0950     NDXSGE=0;
824 0951     NDXSGF=0;
825 0952     END;
826 0953

```

```

: 827      0954 2  ||      IF .XTNPOL NEQ 0
: 828      0955 2  ||      THEN
: 829      0956 2  ||      BEGIN
: 830      0957 2  ||      |
: 831      0958 2  ||      | Clear leftover transaction numbers
: 832      0959 2  ||      |
: 833      0960 2  ||      | FPOOL (XTNPOL):
: 834      0961 2  ||      | XTNCNT = 0:
: 835      0962 2  ||      | XPAGEN = 0:
: 836      0963 2  ||      | XTNLSP = 0:
: 837      0964 2  ||      | XTNLXS = 0:
: 838      0965 2  ||      | XTNSGP = 0:
: 839      0966 2  ||      | XTNTAB = 0:
: 840      0967 2  ||      | END:
: 841      0968 2  ||
: 842      0969 1  ||      END:

```

.TITLE RINIT RUNOFF initialization  
.IDENT \V04-000\

.PSECT \$PLITS\$,NOWRT,NOEXE,2

50	4D	54	2E	4F	4E	52	31	30	30	00000	P.AAA:	.ASCII	\001RNO.TMP\	:			
50	4D	54	2E	4F	4E	52	32	30	30	0000A	P.AAB:	.ASCII	\002RNO.TMP\	:			
				52	4F	52	52	45	24	53	59	53	00014	P.AAC:	.ASCII	\SYSS\$ERROR\	:
				54	55	50	4E	49	24	53	59	53	0001D	P.AAD:	.ASCII	\SYSS\$INPUT\	:
54	55	50	54	55	4F	24	53	59	53	00026	P.AAE:	.ASCII	\SYSS\$OUTPUT\	:			

.PSECT \$OWNS\$,NOEXE,2

```

000A 00000 $IOB$FILE SPEC:
      .WORD 10
01 0E 00002 .BYTE 14, 1
00000000' 00004 .ADDRESS P.AAA
000A 00008 $IOB$FILE SPEC:
      .WORD 10
01 0E 0000A .BYTE 14, 1
00000000' 0000C .ADDRESS P.AAB
0009 00010 $IOB$FILE SPEC:
      .WORD 9
01 0E 00012 .BYTE 14, 1
00000000' 00014 .ADDRESS P.AAC
0009 00018 $IOB$FILE SPEC:
      .WORD 9
01 0E 0001A .BYTE 14, 1
00000000' 0001C .ADDRESS P.AAD
000A 00020 $IOB$FILE SPEC:
      .WORD 10
01 0E 00022 .BYTE 14, 1
00000000' 00024 .ADDRESS P.AAE

```

```

.EXTRN RINTES, MRAA, LSTSTF
.EXTRN PDTSTF, TSFA, FRA
.EXTRN IRA, XMRA, MRA, SBTMRA
.EXTRN TITMRA, SBTTSF, TITTSF
.EXTRN XTSF, KHAR, ATABLE

```

```

.EXTRN TSF, DATTIM, FNESIZ
.EXTRN FNI$IZ, FNCT, FOOREC
.EXTRN GCA, HCT, IF$TK
.EXTRN IRAC, NPAGEN, PAGEN
.EXTRN BRNOOB, FLGT, SCA
.EXTRN SCALIT, FS01, OUTOPT
.EXTRN ECC, FRMSTD, FRMSTK
.EXTRN HLC, HLLIST, HLDSP
.EXTRN SAV$TK, LSTCHR, LSTLCH
.EXTRN LSTRCH, LSTLDD, LSTCNT
.EXTRN LSTSKP, CHLST, LCHLST
.EXTRN RCHLST, LDDLST, NMLST
.EXTRN PDT, PHAN, SKPLST
.EXTRN TTABLE, IOBSTK, RNEIOB
.EXTRN RNAIOB, RNIOB, RNOIOB
.EXTRN FOOIOB, FOTIOB, FOIOB
.EXTRN FOZIOB, TTEIOB, TTIOB
.EXTRN TTOIOB, T$IOB, VRCNT
.EXTRN VRNAME, VRLNG, VRBOOL
.EXTRN VRFFLG, VRTFLG, VRSRC
.EXTRN SAVINI, NATE, RNODAT
.EXTRN SETCAS, XPOSOPEN
.EXTRN XPOSFAILURE

```

.PSECT \$CODE\$,NOWRT,2

```

                                OFFC 0000
5B 00000000G EF 9E 00002
5A 00000000G EF 9E 00009
59 00000000G EF 9E 00010
58 00000000G EF 9E 00017
57 00000000G EF 9E 0001E
56 00000000G EF 9E 00025
   00000000G EF D4 0002C
00000000G EF 9E 00032
00000000G EF D0 0003D
   00000000G EF D4 00048
00000000G EF 9E 0004E
00000000G EF D0 00059
   00000000G EF D4 00064
00000000G EF 9E 0006A
00000000G EF D0 00075
   00000000G EF D4 00080
00000000G EF 9E 00086
00000000G EF D0 00091
   00000000G EF D4 0009C
00000000G EF 9E 000A2
00000000G EF D0 000AD
   00000000G EF D4 000B8
00000000G EF 9E 000BE
00000000G EF D0 000C9
   00000000G EF D4 000D4
00000000G EF 9E 000DA
00000000G EF D0 000E5
   00000000G EF D4 000F0
   00000000GEF40 D4 000F2 1$:
   00000000GEF40 D4 000F9

```

```

.ENTRY RINIT, Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11 : 0385
MOVAB TSF, R11
MOVAB HLC+8, R10
MOVAB PHAN+8, R9
MOVAB SCA, R8
MOVAB FLGT+84, R7
MOVAB GCA+76, R6
CLRL IRA+12 : 0409
MOVAB IRA+16, IRA
MOVL IRA, IRA+4
CLRL MRAA+12 : 0410
MOVAB MRAA+16, MRAA
MOVL MRAA, MRAA+4
CLRL FRA+12 : 0411
MOVAB FRA+16, FRA
MOVL FRA, FRA+4
CLRL FS01+12 : 0412
MOVAB FS01+16, FS01
MOVL FS01, FS01+4
CLRL TITMRA+12 : 0413
MOVAB TITMRA+16, TITMRA
MOVL TITMRA, TITMRA+4
CLRL SBTMRA+12 : 0414
MOVAB SBTMRA+16, SBTMRA
MOVL SBTMRA, SBTMRA+4
CLRL XMRA+12 : 0415
MOVAB XMRA+16, XMRA
MOVL XMRA, XMRA+4
CLRL I : 0420
CLRL TSFAC[I] : 0422
CLRL SBTTSF[I] : 0423

```

			00000000G	EF	40	D4	00100	CLRL	TITTSF[I]	0424	
			00000000G	EF	40	D4	00107	CLRL	XTSF[I]	0425	
	E0		00000000G	50	27	F3	0010E	AOBLEQ	#39, 1, 1\$	0420	
				EF	00	FB	00112	CALLS	#0, SAVINI	0429	
			6B	00000000G	EF	9E	00119	MOVAB	TSFA, TSF	0435	
			50		6B	D0	00120	MOVL	TSF, R0		
		30		A0	20	D0	00123	MOVL	#32, 48(R0)	0436	
			6B	00000000G	EF	9E	00127	MOVAB	SBTTSF, TSF	0437	
			50		6B	D0	0012E	MOVL	TSF, R0		
		30		A0	20	D0	00131	MOVL	#32, 48(R0)	0438	
			6B	00000000G	EF	9E	00135	MOVAB	TITTSF, TSF	0439	
			50		6B	D0	0013C	MOVL	TSF, R0		
		30		A0	20	D0	0013F	MOVL	#32, 48(R0)	0440	
			6B	00000000G	EF	9E	00143	MOVAB	XTSF, TSF	0441	
			50		6B	D0	0014A	MOVL	TSF, R0		
		30		A0	20	D0	0014D	MOVL	#32, 48(R0)	0442	
			00000000G	EF	00000000G	EF	9E	00151	MOVAB	MRAA, MRA	0446
				6B	00000000G	EF	9E	0015C	MOVAB	TSFA, TSF	0447
					00000000G	EF	D4	00163	CLRL	NMLST	0449
					74	A6	7C	00169	CLRQ	GCA+192	0450
		7C		A6	02	D0	0016C	MOVL	#2, GCA+200	0452	
		E8		A6	03	D0	00170	MOVL	#3, GCA+52	0453	
			00000000G	EF	00000000G	EF	D0	00174	MOVL	RNEIOB, RNIOB	0457
			00000000G	EF	00000000G	EF	D0	0017F	MOVL	RNAIOB, RNOIOB	0458
00F4	8F		00	6E	00	2C	0018A	MOVCS	#0, (SP), #0, #244, IOB\$	0459	
							00191				
			00000000G	EF	0301003D	8F	D0	00196	MOVL	#50397245, IOB\$	
			00000000G	EF	020E	8F	B0	001A1	MOVW	#526, IOB\$RESULTANT+2	
00F4	8F		00	6E	00	2C	001AA	MOVCS	#0, (SP), #0, #244, IOB\$	0460	
							001B1				
			00000000G	EF	0301003D	8F	D0	001B6	MOVL	#50397245, IOB\$	
			00000000G	EF	020E	8F	B0	001C1	MOVW	#526, IOB\$RESULTANT+2	
00F4	8F		00	6E	00	2C	001CA	MOVCS	#0, (SP), #0, #244, IOB\$	0461	
							001D1				
			00000000G	EF	0301003D	8F	D0	001D6	MOVL	#50397245, IOB\$	
			00000000G	EF	020E	8F	B0	001E1	MOVW	#526, IOB\$RESULTANT+2	
00F4	8F		00	6E	00	2C	001EA	MOVCS	#0, (SP), #0, #244, IOB\$	0462	
							001F1				
			00000000G	EF	0301003D	8F	D0	001F6	MOVL	#50397245, IOB\$	
			00000000G	EF	020E	8F	B0	00201	MOVW	#526, IOB\$RESULTANT+2	
00F4	8F		00	6E	00	2C	0020A	MOVCS	#0, (SP), #0, #244, IOB\$	0463	
							00211				
			00000000G	EF	0301003D	8F	D0	00216	MOVL	#50397245, IOB\$	
			00000000G	EF	020E	8F	B0	00221	MOVW	#526, IOB\$RESULTANT+2	
00F4	8F		00	6E	00	2C	0022A	MOVCS	#0, (SP), #0, #244, IOB\$	0466	
							00231				
			00000000G	EF	0301003D	8F	D0	00236	MOVL	#50397245, IOB\$	
			00000000G	EF	00000000	EF	9E	00241	MOVAB	\$IOB\$FILE SPEC, IOB\$+4	
			00000000G	EF	020E	8F	B0	0024C	MOVW	#526, IOB\$RESULTANT+2	
			00000000G	EF		01	88	00255	BISB2	#1, IOB\$+48	
00F4	8F		00	6E	00	2C	0025C	MOVCS	#0, (SP), #0, #244, IOB\$	0469	
							00263				
			00000000G	EF	0301003D	8F	D0	00268	MOVL	#50397245, IOB\$	
			00000000G	EF	00000000	EF	9E	00273	MOVAB	\$IOB\$FILE SPEC, IOB\$+4	
			00000000G	EF	020E	8F	B0	0027E	MOVW	#526, IOB\$RESULTANT+2	
			00000000G	EF		01	88	00287	BISB2	#1, IOB\$+48	
			00000000G	EF	00000000G	EF	9E	0028E	MOVAB	F01IOB, F00IOB	0470

00000000G	EF	00000000G	EF	9E	00299	MOVAB	F02I0B, F0TIOB	0471	
00000000G	EF	00000000'	EF	9E	002A4	MOVAB	\$IOB\$FILE_SPEC, IOB\$+4	0505	
00000000G	EF		02	88	002AF	BISB2	#2, IOB\$+46		
00000000G	EF		01	90	002B6	MOVB	#1, IOB\$+44		
		00000000G	EF	9F	002BD	PUSHAB	XPOS\$FAILURE		
			7E	D4	002C3	CLRL	-(SP)		
		00000000G	EF	9F	002C5	PUSHAB	IOB\$		
00000000G	EF		03	FB	002CB	CALLS	#3, XPOS\$OPEN		
00000000G	EF	00000000'	EF	9E	002D2	MOVAB	\$IOB\$FILE_SPEC, IOB\$+4	0512	
00000000G	EF		01	88	002DD	BISB2	#1, IOB\$+46		
00000000G	EF		01	90	002E4	MOVB	#1, IOB\$+44		
		00000000G	EF	9F	002EB	PUSHAB	XPOS\$FAILURE		
			7E	D4	002F1	CLRL	-(SP)		
		00000000G	EF	9F	002F3	PUSHAB	IOB\$		
00000000G	EF		03	FB	002F9	CALLS	#3, XPOS\$OPEN		
00000000G	EF	00000000'	EF	9E	00300	MOVAB	\$IOB\$FILE_SPEC, IOB\$+4	0519	
00000000G	EF		02	88	0030B	BISB2	#2, IOB\$+46		
00000000G	EF		01	90	00312	MOVB	#1, IOB\$+44		
		00000000G	EF	9F	00319	PUSHAB	XPOS\$FAILURE		
			7E	D4	0031F	CLRL	-(SP)		
		00000000G	EF	9F	00321	PUSHAB	IOB\$		
00000000G	EF		03	FB	00327	CALLS	#3, XPOS\$OPEN		
00000000G	FF		02	D0	0032E	MOVL	#2, @HCT+20	0523	
00000000G	FF		01	D0	00335	MOVL	#1, @HCT+8	0524	
00000000G	FF		01	D0	0033C	MOVL	#1, @HCT+12	0525	
00000000G	FF		01	D0	00343	MOVL	#1, @HCT+44	0526	
		00000000G	FF	D4	0034A	CLRL	@HCT+24	0527	
		00000000G	EF	7C	00350	CLRQ	HCT+28	0533	
		00000000G	EF	7C	00356	CLRQ	HCT+36	0535	
		00000000G	EF	D4	0035C	CLRL	HCT+16	0537	
			50	D4	00362	CLRL	I	0542	
		00000000G	EF	40	D4	00364	CLRL	FNCT[I]	0543
F5			0B	F3	0036B	AOBLEQ	#11, I, 2\$		
			50	D4	0036F	CLRL	I	0545	
		00000000G	EF	40	D4	00371	CLRL	FNESIZ[I]	0547
		00000000G	EF	40	D4	00378	CLRL	FNISIZ[I]	0548
EE			13	F3	0037F	AOBLEQ	#19, I, 3\$	0545	
			50	D4	00383	CLRL	I	0554	
		00000000G	EF	40	D4	00385	CLRL	FORECC[I]	0555
F5			03	F3	0038C	AOBLEQ	#3, I, 4\$		
		00000000G	EF	D4	00390	CLRL	VRTCNT	0560	
00000000G	EF		20	D0	00396	MOVL	#32, VRTFLG	0561	
00000000G	EF		20	D0	0039D	MOVL	#32, VRFFLG	0562	
00000000G	EF		01	D0	003A4	MOVL	#1, VRBOOL	0563	
		00000000G	EF	D4	003AB	CLRL	IFSTK+16	0568	
00000000G	EF		01	7D	003B1	MOVQ	#1, IFSTK+8	0569	
		00000000G	EF	7C	003B8	CLRQ	IFSTK	0567	
00000000G	EF		20	D0	003BE	MOVL	#32, IFSTK+28	0572	
00000000G	EF	00000000G	EF	9E	003C5	MOVAB	CHLST, LSTCHR	0576	
		00000000G	FF	D4	003D0	CLRL	@LSTCHR	0577	
00000000G	EF	00000000G	EF	9E	003D6	MOVAB	LSTSTF, LSTCNT	0581	
00000000G	EF	00000000G	EF	9E	003E1	MOVAB	LCHLST, LSTLCH	0585	
00000000G	EF	00000000G	EF	9E	003EC	MOVAB	RCHLST, LSTRCH	0586	
00000000G	EF	00000000G	EF	9E	003F7	MOVAB	LDDLST, LSTLDD	0587	
		00000000G	FF	D4	00402	CLRL	@LSTLCH	0588	
00000000G	FF		2E	D0	00408	MOVL	#46, @LSTRCH	0589	
		00000000G	FF	D4	0040F	CLRL	@LSTLDD	0590	

		50	00000000G	EF	D0	00415		MOVL	LSTCNT, R0		0591	
		60		0F	00	0041C		MOVL	#15, (R0)			
	04	A0		01	7D	0041F		MOVQ	#1, 4(R0)		0592	
	00000000G	EF	00000000G	EF	9E	00423		MOVAB	SKPLST, LSTSKP		0597	
	00000000G	FF		01	D0	0042E		MOVL	#1, @LSTSKP		0598	
	00000000G	EF		06	D0	00435		MOVL	#6, HLLIST		0602	
	00000000G	EF		01	D0	0043C		MOVL	#1, HLLIST+4		0603	
		51	00000000G	EF	D0	00443		MOVL	HLLIST, R1		0608	
				50	D4	0044A		CLRL	I			
				0E	11	0044C		BRB	6\$			
			00000000G	EF	40	D4	0044E	5\$:	CLRL	HLDSP-4[I]	0610	
			00000000G	EF	40	D4	00455		CLRL	HLLIST+4[I]	0611	
EE		50		51	F3	0045C	6\$:	AOBLEQ	R1, I, 5\$		0608	
			00000000G	EF	7C	00460		CLRQ	SAVSTK+8		0622	
			00000000G	EF	7C	00466		CLRQ	SAVSTK		0619	
				01	DD	0046C		PUSHL	#1		0628	
				0B	DD	0046E		PUSHL	#11			
				7E	D4	00470		CLRL	-(SP)			
	00000000V	EF		03	FB	00472		CALLS	#3, ECCINI			
		6A		03	7D	00479		MOVQ	#3, HLC+8		0636	
	04	AA		04	88	0047C		BISB2	#4, HLC+12		0633	
	F8	AA		01	D0	00480		MOVL	#1, HLC		0634	
	FC	AA		06	D0	00484		MOVL	#6, HLC+4		0635	
	08	AA		01	D0	00488		MOVL	#1, HLC+16		0637	
	0C	AA		07	D0	0048C		MOVL	#7, HLC+20		0638	
	10	AA		03	D0	00490		MOVL	#3, HLC+24		0639	
	18	AA		02	D0	00494		MOVL	#2, HLC+32		0640	
	14	AA	63	8F	9A	00498		MOVZBL	#99, HLC+28		0641	
	1C	AA	63	8F	9A	0049D		MOVZBL	#99, HLC+36		0642	
	00000000G	EF		28	D0	004A2		MOVL	#40, TTABLE		0646	
	00000000G	EF		08	D0	004A9		MOVL	#8, TTABLE+8		0647	
		50		02	D0	004B0		MOVL	#2, I		0649	
00000000G	EF	40	00000000G	EF	40	08	C1	004B3	7\$:	ADDL3	#8, TTABLE[I], TTABLE+4[I]	0650
EE		50		28	F3	004C1		AOBLEQ	#40, I, 7\$			
			00000000G	EF	28	D0	004C5		MOVL	#40, TTABLE+4		0652
				52	D4	004CC		CLRL	I		0660	
				52	DD	004CE	8\$:	PUSHL	I		0661	
	00000000G	EF		01	FB	004D0		CALLS	#1, NATE			
EF		52	000000FF	8F	F3	004D7		AOBLEQ	#255, I, 8\$			
				7E	D4	004DF		CLRL	-(SP)		0666	
	00000000G	EF		01	FB	004E1		CALLS	#1, SETCAS			
			AC	A7	D4	004E8		CLRL	FLGT		0670	
	B0	A7		01	D0	004EB		MOVL	#1, FLGT+4		0671	
	B4	A7		01	D0	004EF		MOVL	#1, FLGT+8		0672	
	B8	A7		01	D0	004F3		MOVL	#1, FLGT+12		0673	
	BC	A7		01	D0	004F7		MOVL	#1, FLGT+16		0674	
	C0	A7		01	7D	004FB		MOVQ	#1, FLGT+20		0675	
	C8	A7		01	7D	004FF		MOVQ	#1, FLGT+28		0677	
	D0	A7		01	7D	00503		MOVQ	#1, FLGT+36		0679	
			D8	A7	7C	00507		CLRQ	FLGT+44		0681	
	E0	A7		01	D0	0050A		MOVL	#1, FLGT+52		0683	
	E4	A7		01	7D	0050E		MOVQ	#1, FLGT+56		0684	
			EC	A7	D4	00512		CLRL	FLGT+64		0686	
	F4	A7		24	D0	00515		MOVL	#36, FLGT+72		0695	
	F8	A7		21	D0	00519		MOVL	#33, FLGT+76		0696	
	FC	A7		2E	D0	0051D		MOVL	#46, FLGT+80		0697	
		67	SF	8F	9A	00521		MOVZBL	#95, FLGT+84		0698	

04	A7	5E	8F	9A	00525	MOVZBL	#94, FLGT+88	0699
08	A7	5C	8F	9A	0052A	MOVZBL	#92, FLGT+92	0700
0C	A7		3C	D0	0052F	MOVL	#60, FLGT+96	0701
10	A7		26	D0	00533	MOVL	#38, FLGT+100	0702
14	A7		2A	D0	00537	MOVL	#42, FLGT+104	0703
18	A7		23	D0	0053B	MOVL	#35, FLGT+108	0704
1C	A7		3E	D0	0053F	MOVL	#62, FLGT+112	0705
20	A7		3D	D0	00543	MOVL	#61, FLGT+116	0706
24	A7		25	D0	00547	MOVL	#37, FLGT+120	0707
28	A7		3E	D0	0054B	MOVL	#62, FLGT+124	0708
2C	A7		21	D0	0054F	MOVL	#33, FLGT+128	0709
30	A7		2B	D0	00553	MOVL	#43, FLGT+132	0710
34	A7	7C	8F	9A	00557	MOVZBL	#124, FLGT+136	0711
50	00000000G		EF	9E	0055C	MOVAB	ATABLE, R0	0720
00	B740		03	90	00563	MOVB	#3, @FLGT+84[R0]	
50	00000000G		EF	9E	00568	MOVAB	ATABLE, R0	0721
04	B740		03	90	0056F	MOVB	#3, @FLGT+88[R0]	
50	00000000G		EF	9E	00574	MOVAB	ATABLE, R0	0722
08	B740		03	90	0057B	MOVB	#3, @FLGT+92[R0]	
50	00000000G		EF	9E	00580	MOVAB	ATABLE, R0	0723
10	B740		03	90	00587	MOVB	#3, @FLGT+100[R0]	
50	00000000G		EF	9E	0058C	MOVAB	ATABLE, R0	0724
18	B740		03	90	00593	MOVB	#3, @FLGT+108[R0]	
00000000G	EF		01	D0	00598	MOVL	#1, IRAC+12	0728
	00000000G		EF	D4	0059F	CLRL	IRAC+8	0729
00000000G	EF		01	8A	005A5	BICB2	#1, IRAC	0730
00000000G	EF		20	D0	005AC	MOVL	#32, IRAC+24	0731
			50	D4	005B3	CLRL	I	0746
			6840	D4	005B5	CLRL	SCA[I]	
F9	50		18	F3	005B8	AOBLEQ	#24, I, 9\$	6E
		6C	B8	D4	005BC	CLRL	@SCA+108	20
		00B4	D8	D4	005BF	CLRL	@SCA+132	
	50		25	D0	005C3	MOVL	#37, I	00
			6840	D4	005C6	CLRL	SCA[I]	65
F5	50	0000005F	8F	F3	005C9	AOBLEQ	#95, I, 10\$	53
00D0	C8		01	D0	005D1	MOVL	#1, SCA+208	
78	B8	46	8F	9A	005D6	MOVZBL	#70, @SCA+120	6C
		74	B8	D4	005DB	CLRL	@SCA+116	6E
7C	B8		01	D0	005DE	MOVL	#1, @SCA+124	
0094	C8		01	D0	005E2	MOVL	#1, SCA+148	64
68	B8		01	D0	005E7	MOVL	#1, @SCA+104	22
64	B8		01	D0	005EB	MOVL	#1, @SCA+100	
70	B8		01	D0	005EF	MOVL	#1, @SCA+112	
0080	D8		01	D0	005F3	MOVL	#1, @SCA+128	
0090	D8		01	D0	005F8	MOVL	#1, @SCA+144	6C
00A8	C8		1F	88	005FD	BISB2	#31, SCA+168	2D
008C	D8		01	D0	00602	MOVL	#1, @SCA+140	
00D4	C8		01	D0	00607	MOVL	#1, SCA+212	69
00B4	C8		01	7D	0060C	MOVQ	#1, SCA+180	20
0088	D8	7C	8F	9A	00611	MOVZBL	#124, @SCA+136	
00F8	C8	00000000G	FF	D0	00617	MOVL	@MRA, SCA+248	69
0118	C8	00G	8F	9A	00620	MOVZBL	#RINTES, SCA+280	6F
00000000G	EF	5F	8F	9A	00626	MOVZBL	#95, OUTOPT	
		00000000G	EF	7C	0062E	CLRQ	OUTOPT+4	0780
00000000G	EF		01	D0	00634	MOVL	#1, OUTOPT+16	0781
		00000000G	EF	D4	0063B	CLRL	OUTOPT+12	0784
		00000000G	EF	9F	00641	PUSHAB	DATTIM	0785
								0789

66  
69  
00  
6E  
20  
00  
65  
53  
6C  
6E  
64  
22  
6C  
2D  
69  
20  
69  
6F  
75  
6E  
65



00000000G	EF		01	FB	00647	CALLS	#1, RNODAT			
BC	B6		01	D0	0064E	MOVL	#1, @GCA+8	0793		
		B8	B6	D4	00652	CLRL	@GCA+4	0794	6F	
		CO	B6	D4	00655	CLRL	@GCA+12	0795	65	
C4	B6		01	D0	00658	MOVL	#1, @GCA+16	0796	70	
		0080	D6	D4	0065C	CLRL	@GCA+204	0797		
2C	A6		C1	D0	00660	MOVL	#1, GCA+120	0798	74	
C8	A6		01	8A	00664	BICB2	#1, GCA+20	0799		
CC	A6		01	8A	00668	BICB2	#1, GCA+24	0800	71	
D0	A6		01	8A	0066C	BICB2	#1, GCA+28	0801	22	
D4	A6		01	8A	00670	BICB2	#1, GCA+32	0802		
	66		01	88	00674	BISB2	#1, GCA+76	0803		
	66		08	8A	00677	BICB2	#8, GCA+76	0804	63	
0085	C6		04	8A	0067A	BICB2	#4, GCA+209	0805	74	
28	A6		0F	8A	0067F	BICB2	#15, GCA+116	0809		
F8	A6		0F	88	00683	BISB2	#15, GCA+68	0813	7A	
30	A6		07	8A	00687	BICB2	#7, GCA+124	0816	45	
3C	A6	96	8F	9A	00688	MOVZBL	#150, GCA+136	0817		
40	B6	46	8F	9A	00690	MOVZBL	#70, @GCA+140	0818		
48	A6		08	D0	00695	MOVL	#8, GCA+148	0820	6F	
4C	A6		04	D0	00699	MOVL	#4, GCA+152	0822	4E	
50	A6		01	8A	0069D	BICB2	#1, GCA+156	0823		
5C	A6		01	D0	006A1	MOVL	#1, GCA+168	0824	65	
60	A6		01	D0	006A5	MOVL	#1, GCA+172	0825	6E	
64	A6		01	D0	006A9	MOVL	#1, GCA+176	0826		
08	B6		03	D0	006AD	MOVL	#3, @GCA+84	0831		
6C	A6		0A	7D	006B1	MOVQ	#10, GCA+184	0834	69	
		0C	A6	7C	006B5	CLRQ	GCA+88	0836	22	
		1C	A6	D4	006B8	CLRL	GCA+104	0839		
		14	A6	7C	006BB	CLRQ	GCA+96	0838		
		0094	C6	7C	006BE	CLRQ	GCA+224	0853	69	
0099	C6	3F0000FF	8F	CA	006C2	BICL2	#1056964863, GCA+229	0849	2E	
			F0	A6	D4	006CB	CLRL	GCA+60	0851	
38	B6		8F	9A	006CE	MOVZBL	#70, @GCA+132	0852	72	
			B6	D4	006D3	CLRL	@GCA+128	0854		
	66		14	8A	006D6	BICB2	#20, GCA+76	0856	69	
		20	A6	7C	006D9	CLRQ	GCA+108	0857	52	
0084	C6		01	8A	006DC	BICB2	#1, GCA+208	0859		
	69	42	8F	9A	006E1	MOVZBL	#66, PHAN+8	0869	6C	
1C	A9	42	8F	9A	006E5	MOVZBL	#66, PHAN+36	0872		
FC	B9		3A	D0	006EA	MOVL	#58, @PHAN+4	0877	20	
F8	A9		01	D0	006EE	MOVL	#1, PHAN	0878	61	
		04	A9	7C	006F2	CLRQ	PHAN+12	0879		
10	A9		01	D0	006F5	MOVL	#1, PHAN+24	0881		
		14	B9	D4	006F9	CLRL	@PHAN+28	0882	64	
		18	A9	D4	006FC	CLRL	PHAN+32	0883	2E	
20	B9		01	D0	006FF	MOVL	#1, @PHAN+40	0884		
		0C	A9	D4	00703	CLRL	PHAN+20	0886		
24	A9		01	7D	00706	MOVQ	#1, PHAN+44	0885	61	
		2C	A9	7C	0070A	CLRQ	PHAN+52	0888	69	
		34	A9	7C	0070D	CLRQ	PHAN+60	0890		
		3C	A9	D4	00710	CLRL	PHAN+68	0892	73	
			50	D4	00713	CLRL	I	0901	00	
		00000000GEF	40	D4	00715	CLRL	PAGEN[1]	0902		
F5	50		03	F3	0071C	AOBLEQ	#3, I, 11\$			
00000000G	EF		01	D0	00720	MOVL	#1, PAGEN+8	0906		
00000000G	EF		01	B0	00727	MOVW	#1, PAGEN+14	0907	67	

11\$:

RINIT  
V04-000

RUNOFF initialization  
RINIT -- body of main routine

G 1  
16-Sep-1984 01:32:39  
14-Sep-1984 13:07:51

VAX-11 Bliss-32 V4.0-742  
DISK\$VMSMASTER:[RUNOFF.SRC]RINIT.BLI;1 (4) Page 24

00000000G	EF	04	00000000G	EF	FO	8F	8A	0072E	BICB2	#240, PAGEN	:	0912
				04		04	FO	00736	INSV	#4, #4, #4, PAGEN+13	:	0913
00000000G	EF	04	00000000G	EF	FO	8F	8A	0073F	BICB2	#240, PAGEN+12	:	0914
00000000G	EF	04		00		02	FO	00747	INSV	#2, #0, #4, PAGEN+13	:	0915
00000000G	EF	04		00		02	FO	00750	INSV	#2, #0, #4, PAGEN+12	:	0916
						5C	D4	00759	CLRL	I	:	0926
			00000000G	EF	40	00000000G	EF	40	DO	0075B 12\$:	:	0927
				50		03	F3	00768	AOBLEQ	#3, I, 12\$	:	0931
			00000000G	EF		02	DO	0076C	MOVL	#2, NPAGEN+8	:	0932
			00000000G	EF		02	BO	00773	MOVW	#2, NPAGEN+14	:	0936
			00000000G	EF	00000000G	EF	9E	0077A	MOVAB	PDT, R0	:	0937
				50	00000000G	EF	DO	00785	MOVL	PDT, R0	:	0939
				08	AO	02	DO	0078C	MOVL	#2, 8(R0)	:	0941
				04	AO	01	DO	00790	MOVL	#1, 4(R0)	:	0969
				60		05	DO	00794	MOVL	#5, (R0)	:	
						04	DO	00797	RET		:	

; Routine Size: 1944 bytes, Routine Base: \$CODE\$ + 0000

; 843 0970 1

RNI  
V04  
61  
00  
72  
69  
64  
65  
77  
74  
27  
65  
63  
00  
6F  
61  
22  
6F  
6E  
63  
73  
72  
20  
65  
6F  
6F  
6E  
6C  
41  
78  
6C  
27

845  
846  
847  
848  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
860  
861  
862  
863  
864  
865  
866  
867  
868  
869  
870  
871  
872  
873  
874  
875  
876  
877  
878  
879  
880  
881  
882  
883  
884  
885  
886  
887  
888  
889  
890  
891  
892  
893  
894  
895  
896  
897  
898  
899  
900  
901

```

0971 1 %SBTTL 'ECCINI -- initialize ECC structure to default values'
0972 1 GLOBAL ROUTINE eccini (first, last, counters) : NOVALUE =
0973 1
0974 1 ++
0975 1
0976 1 FUNCTIONAL DESCRIPTION:
0977 1
0978 1     ECCINI initializes one or more parts of the ECC structure.
0979 1
0980 1 FORMAL PARAMETERS:
0981 1
0982 1     FIRST is the offset of the first $ECC_BLOCK to initialize
0983 1
0984 1     LAST is the offset of the last $ECC_BLOCK to initialize
0985 1
0986 1     COUNTERS tells whether or not to zero the entity counters.
0987 1
0988 1     For example, to initialize all the .STYLE { example :
0989 1     figure ; table } data including counters:
0990 1
0991 1         ECCINI (EXAMP_OFFSET, TABLE_OFFSET, TRUE)
0992 1
0993 1     To initialize all the .STYLE HEADER data but leave counters
0994 1     intact:
0995 1
0996 1         ECCINI (HL1_OFFSET, HL6_OFFSET, FALSE)
0997 1
0998 1 IMPLICIT INPUTS:      None
0999 1
1000 1 IMPLICIT OUTPUTS:
1001 1
1002 1     Parts of the ECC are initialized.
1003 1
1004 1 ROUTINE VALUE:
1005 1 COMPLETION CODES:    None
1006 1
1007 1 SIDE EFFECTS:       None
1008 1
1009 1 --
1010 2 BEGIN
1011 2 OWN
1012 2     FS_ALLOCATE (POST_STRING, 5);           ! For default ":" post string.
1013 2
1014 2     FS_INIT (POST_STRING);
1015 2     FS_WCHAR (POST_STRING, %C':');         ! Insert a ":" after the counter..
1016 2
1017 2 IF .FIRST LSS 0 OR .LAST GTR MAX_ECC_OFFSET
1018 2 THEN
1019 2     RETURN;
1020 2
1021 2 INCR I FROM .FIRST TO .LAST DO
1022 2     BEGIN
1023 2         BIND SB = ECC [I, 0,0,0,0] : $ECC_BLOCK;
1024 2
1025 2         ! Zero all the Option bits, the effect is:
1026 2         ! Numbered headers; no bold; no underline; no break before caption;
1027 2         ! For CHAPTER/APPENDIX: this means,

```

RN  
VO  
44  
65  
60  
65  
60  
61  
63  
60  
3A  
70  
74  
72  
53  
70  
74  
6F  
20  
25  
20  
72  
65  
64  
72  
20  
2E  
6F

Line	Code	Comment	Line
902	1028	! no insertion of page number into the TOC, don't start a new page,	61
903	1029	! and don't necessarily start on an odd-numbered page.	69
904	1030		
905	1031	SB [ECC\$V_OPTIONS] = 0;	64
906	1032		
907	1033	IF (.I GEQ HL3_OFFSET) AND (.I LSS CHAP_OFFSET) ! Run-in HL3 --> HL6.	45
908	1034	THEN	69
909	1035	SB [ECC\$H_RUNIN] = TRUE;	
910	1036		00
911	1037	SB [ECC\$H_CASE] =	
912	1038	(IF .I LEQ TABLE_OFFSET	! Case rules for header:
913	1039	THEN	
914	1040	ECC\$K_LEAVECASE	! No case change for <entities>.
915	1041	ELSE	
916	1042	IF .I EQL HL1_OFFSET	
917	1043	THEN	
918	1044	ECC\$K_ALLCAP	! All cap for HL1,
919	1045	ELSE	
920	1046	ECC\$K_FIRSTCAP);	! Initial cap for HL2 --> HL6.
921	1047		
922	1048	SB [ECC\$H_POSITION] = ECC\$K_LEFT;	! Position header flush-left.
923	1049		58
924	1050	SB [ECC\$H_AFTER] = 1;	! Skip 1 line after header.
925	1051		
926	1052	SB [ECC\$H_BEFORE] =	! Skip lines before header:
927	1053	(IF .I LEQ TABLE_OFFSET	
928	1054	THEN 1	! 1 for <entities>.
929	1055	ELSE 3);	! 3 for HL's.
930	1056		20
931	1057	SB [ECC\$H_SPACES] = 2;	! 2 spaces between counter and caption
932	1058		00
933	1059	IF .I EQL HCOLL_OFFSET	
934	1060	THEN	
935	1061	SB [ECC\$H_TESTPAGE] = 7	! Testpage for HL's.
936	1062	ELSE	
937	1063	SB [ECC\$H_TESTPAGE] = 6;	! Start new page if not n lines remaining:
938	1064		! 6 = 4 + <before> + <after> for entities,
939	1065		6E
940	1066		65
941	1067	IF .I GEQ CHAP_OFFSET	
942	1068	THEN	
943	1069	BEGIN	! CHAPTER/APPENDIX rules
944	1070	SB [ECC\$V_PAGE] = TRUE;	! Start a new page before CHAP/APP.
945	1071	SB [ECC\$H_BETWEEN] = 1;	! Skip one line between counter and caption (CHAPTER).
946	1072	SE [ECC\$V_BREAK] = TRUE;	! Put a break between counter & caption.
947	1073	SB [ECC\$H_POSITION] = ECC\$K_CENTER;	! Position CHAPTERS CENTER.
948	1074	SB [ECC\$H_BEFORE] = 12;	! Skip lines before the word chapter:
949	1075	SB [ECC\$H_AFTER] = 3;	! Skip 3 lines after the word chapter.
950	1076	SB [ECC\$V_STARTODD] = FALSE;	! Don't care what page to start on.
951	1077	SB [ECC\$V_TOCPAGE] = FALSE;	! Don't put the page number in TOC.
952	1078	SB [ECC\$H_TESTPAGE] = 17;	! Start new page if not n lines remaining:
953	1079	SB [ECC\$H_CASE] = ECC\$K_ALLCAP;	! Allcap for chapters.
954	1080	END;	
955	1081		65
956	1082		52
957	1083	!+ Zero the counters initially (when called from RINIT); do not zero them	6F
958	1084	! during processing (when called from STYLE).	6E



				66	D4	00009	CLRL	POST_STRING+12	1014
	F4	A6	04	A6	9E	0000B	MOVAB	POST_STRING+16, POST_STRING	
	F8	A6	F4	A6	D0	00010	MOVL	POST_STRING, POST_STRING+4	
	F8	B6		3A	90	00015	MOVB	#58, @POST_STRING+4	1015
			F8	A6	D6	00019	INCL	POST_STRING+4	
				66	D6	0001C	INCL	POST_STRING+12	
			04	AC	D5	0001E	TSTL	FIRST	1017
				01	18	00021	BGEQ	1\$	
				04	00023		RET		
		0B	08	AC	D1	00024	1\$:	C MPL LAST, #11	
				01	15	00028		BLEQ 2\$	
				04	0002A			RET	
50	04	AC		01	C3	0002B	2\$:	SUBL3 #1, FIRST, I	1086
				00FD	31	00030		BRW 16\$	
51		50		24	C5	00033	3\$:	MULL3 #36, I, R1	1023
		51	00000000GEF	41	9E	00037		MOVAB ECC[R1], R1	
				61	B4	0003F		CLRW (R1)	1031
		06		50	D1	00041		C MPL I, #6	1033
				09	19	00044		BLSS 4\$	
		0A		50	D1	00046		C MPL I, #10	
				04	18	00049		BGEQ 4\$	
	02	A1		01	90	0004B		MOVB #1, 2(R1)	1035
				55	D4	0004F	4\$:	CLRL R5	1038
		02		50	D1	00051		C MPL I, #2	
				06	14	00054		BGTR 5\$	
				55	D6	00056		INCL R5	
				52	D4	00058		CLRL R2	
				0D	11	0005A		BRB 7\$	
		04		50	D1	0005C	5\$:	C MPL I, #4	1042
				05	12	0005F		BNEQ 6\$	
		52		02	D0	00061		MOVL #2, R2	
				03	11	00064		BRB 7\$	
		52		01	D0	00066	6\$:	MOVL #1, R2	
	03	A1		52	90	00069	7\$:	MOVB R2, 3(R1)	1038
		53	04	A1	9E	0006D		MOVAB 4(R1), R3	1048
				63	94	00071		CLRB (R3)	
	01	A3		01	B0	00073		MOVW #1, 1(R3)	1050
		05		55	E9	00077		BLBC R5, 8\$	1053
		52		01	D0	0007A		MOVL #1, R2	
				03	11	0007D		BRB 9\$	
		52		03	D0	0007F	8\$:	MOVL #3, R2	
	08	63	10	18	52	F0	9\$:	INSV R2, #24, #16, (R3)	
		A1	10	18	02	F0		INSV #2, #24, #16, 8(R1)	1057
				54	D4	0008D		CLRL R4	1059
				50	D1	0008F		C MPL I, #3	
		03		0C	12	00092		BNEQ 10\$	
				54	D6	00094		INCL R4	
	01	52	0C	A1	9E	00096		MOVAB 12(R1), R2	1061
		A2		07	B0	0009A		MOVW #7, 1(R2)	
				08	11	0009E		BRB 11\$	
	01	52	0C	A1	9E	000A0	10\$:	MOVAB 12(R1), R2	1063
		A2		06	B0	000A4		MOVW #6, 1(R2)	
		0A		50	D1	000A8	11\$:	C MPL I, #10	1066
				22	19	000AB		BLSS 12\$	
		61		20	88	000AD		BISB2 #32, (R1)	1069
	09	A1		01	B0	000B0		MOVW #1, 9(R1)	1070
		61		08	88	000B4		BISB2 #8, (R1)	1071

63	10	01	63	02	90	000B7	MOVB	#2, (R3)	1072
			18	0C	F0	000BA	INSV	#12, #24, #16, (R3)	1073
			A3	03	B0	000BF	MOVW	#3, 1(R3)	1074
			61	50	8F	8A 000C3	BICB2	#80, (R1)	1076
			A2		11	B0 000C7	MOVW	#17, 1(R2)	1077
			A1	03	02	90 000CB	MOVB	#2, 3(R1)	1078
62	20		35	0C	AC	E9 000CF	BLBC	COUNTERS, 15\$	1086
			18		00	F0 000D3	INSV	#0, #24, #32, (R2)	1089
			0B		50	D1 000D8	CPL	I, #11	1091
					08	13 000DB	BEQL	13\$	
10	A1	10	18		00	F0 000DD	INSV	#0, #24, #16, 16(R1)	1093
					06	11 000E3	BRB	14\$	
10	A1	10	18		02	F0 000E5	INSV	#2, #24, #16, 16(R1)	1096
				15	A1	B4 000EB	CLRW	21(R1)	1099
14	A1	20	18		00	F0 000EE	INSV	#0, #24, #32, 20(R1)	1100
					50	D5 000F4	TSTL	I	1102
					10	19 000F6	BLSS	15\$	
			0D		55	E9 000F8	BLBC	R5, 15\$	
18	A1	10	18		66	F0 000FB	INSV	POST_STRING+12, #24, #16, 24(R1)	1105
1C	A1	20	08	F4	A6	F0 00101	INSV	POST_STRING, #8, #32, 28(R1)	1106
			25		54	E9 00108	BLBC	R4, T6\$	1111
			61	FFFF0000	8F	D0 0010B	MOVL	#-65536, (R1)	1114
			63		01	8E 00112	MNEGB	#1, (R3)	1117
			A3	01	01	00115	MNEGW	#1, 1(R3)	1118
			18	FFFFFFFF	8F	F0 00119	INSV	#-1, #24, #16, (R3)	1119
0B	A1	10	18	FFFFFFFF	8F	F0 00122	INSV	#-1, #24, #16, 8(R1)	1120
			A2	01	01	AE 0012C	MNEGW	#1, 1(R2)	1121
FEFC	50		01	08	AC	F1 00130	ACBL	LAST, #1, 1, 3\$	1021
					04	00137	RET		1126

: Routine Size: 312 bytes, Routine Base: \$CODE\$ + 0798

: 1001 1127 1  
: 1002 1128 1 END  
: 1003 1129 0 ELUDOM

: End of module

PSECT SUMMARY

Name	Bytes	Attributes
\$PLITS	48	NOVEC, NOWRT, RD, NOEXE, NOSHR, LCL, REL, CON, NOPIC, ALIGN(2)
\$OWNS	61	NOVEC, WRT, RD, NOEXE, NOSHR, LCL, REL, CON, NOPIC, ALIGN(2)
\$CODE\$	2256	NOVEC, NOWRT, RD, EXE, NOSHR, LCL, REL, CON, NOPIC, ALIGN(2)

Library Statistics

File	----- Symbols -----		Pages Mapped	Processing Time
	Total	Loaded Percent		

RINIT  
V04-000

RUNOFF initialization  
ECCINI -- initialize ECC structure to default v

M 1  
16-Sep-1984 01:32:39  
14-Sep-1984 13:07:51

VAX-11 Bliss-32 V4.0-742  
DISK\$VMSMASTER:[RUNOFF.SRC]RINIT.BLI;1 (5)  
Page 30

RN

```

:
:  $255$DUA28:[SYSLIB]XPORT.L32;1          590      106      17      252      00:00.1
:  _$255$DUA28:[RUNOFF.SRC]DSRLIB.L32;1    1248     302     24      86      00:00.3

```

COMMAND QUALIFIERS

BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LISS:RINIT/OBJ=OBJ\$:RINIT MSRCS:RINIT/UPDATE=(ENHS:RINIT)

```

: Size:          2256 code + 109 data bytes
: Run Time:      01:08.0
: Elapsed Time: 02:16.3
: Lines/CPU Min: 996
: Lexemes/CPU-Min: 29864
: Memory Used:  494 pages
: Compilation Complete

```





A grid of 20 columns and 15 rows of terminal screens. Each screen displays a different VAX/VMS utility or command-line interface. The screens are arranged in a regular grid pattern. Some screens are more prominent than others, showing larger text or distinct headers. The overall appearance is that of a multi-terminal session or a printed representation of a multi-terminal environment.

RNODEF  
LIS

RSDEMI  
LIS

RTERM  
LIS

RNOUMS  
LIS

RSKIP  
LIS

RUNOFF  
LIS

RNODAT  
LIS

RNOURS  
LIS

SAURES  
LIS

RNFERM  
LIS