



```

NN      NN  DDDDDDD  XX      XX  XX      XX  TTTTTTTTTT  NN      NN
NN      NN  DDDDDDD  XX      XX  XX      XX  TTTTTTTTTT  NN      NN
NN      NN  DD      DD  XX      XX  XX      XX  TT      NN      NN
NN      NN  DD      DD  XX      XX  XX      XX  TT      NN      NN
NNNN    NN  DD      DD  XX  XX  XX  XX  TT      NNNN    NN
NNNN    NN  DD      DD  XX  XX  XX  XX  TT      NNNN    NN
NN  NN  NN  DD      DD  XX      XX  XX      XX  TT      NN  NN  NN
NN  NN  NN  DD      DD  XX      XX  XX      XX  TT      NN  NN  NN
NN      NN  DD      DD  XX      XX  XX      XX  TT      NN      NN
NN      NN  DD      DD  XX      XX  XX      XX  TT      NN      NN
NN      NN  DDDDDDD  XX      XX  XX      XX  TT      NN      NN
NN      NN  DDDDDDD  XX      XX  XX      XX  TT      NN      NN

```

```

LL      IIIIII  SSSSSSSS
LL      IIIIII  SSSSSSSS
LL      II     SS
LL      II     SS
LL      II     SS
LL      II     SS
LL      II     SSSSSS
LL      II     SSSSSS
LL      II     SS
LL      II     SS
LL      II     SS
LL      II     SS
LLLLLLLLLLLL IIIIII  SSSSSSSS
LLLLLLLLLLLL IIIIII  SSSSSSSS

```

```

1 0001 0 MODULE NDXXTN (IDENT = 'V04-000'
2 0002 0 %BLISS32 [, ADDRESSING_MODE (EXTERNAL = LONG_RELATIVE, NONEXTERNAL = LONG_RELATIVE)]
3 0003 0 ) =
4 0004 1 BEGIN
5 0005 1
6 0006 1 *****
7 0007 1 *
8 0008 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY *
9 0009 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS. *
10 0010 1 * ALL RIGHTS RESERVED. *
11 0011 1 *
12 0012 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
13 0013 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE *
14 0014 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER *
15 0015 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
16 0016 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY *
17 0017 1 * TRANSFERRED. *
18 0018 1 *
19 0019 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE *
20 0020 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT *
21 0021 1 * CORPORATION. *
22 0022 1 *
23 0023 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS *
24 0024 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL. *
25 0025 1 *
26 0026 1 *
27 0027 1 *****
28 0028 1
29 0029 1 ++
30 0030 1 FACILITY:
31 0031 1 DSR (Digital Standard RUNOFF) /DSRPLUS DSRINDEX/INDEX Utility
32 0032 1
33 0033 1 ABSTRACT: Routines for processing transaction numbers.
34 0034 1
35 0035 1
36 0036 1 ENVIRONMENT: Transportable
37 0037 1
38 0038 1 AUTHOR: RWF
39 0039 1
40 0040 1 CREATION DATE: January, 1979
41 0041 1
42 0042 1 MODIFIED BY:
43 0043 1
44 0044 1 004 JPK00015 04-feb-1983
45 0045 1 Cleaned up module names, modified revision history to
46 0046 1 conform with established standards. Updated copyright dates.
47 0047 1
48 0048 1 003 JPK00012 24-Jan-1983
49 0049 1 Modified NDXVMSMSG.MSG to define error messages for both
50 0050 1 DSRINDEX and INDEX.
51 0051 1 Added require of NDXVMSREQ.R32 to NDXOUT, NDXFMT, NDXDAT,
52 0052 1 INDEX, NDXMSG, NDXXTN, NDXTMS, NDXVMS and NDXPAG for BLISS32.
53 0053 1 Since this file defines the error message literals,
54 0054 1 the EXTERNAL REFERENCES for the error message literals
55 0055 1 have been removed.
56 0056 1
57 0057 1 002 JPK00008 19-Nov-1982

```

NDXXTN  
V04-000

N 1  
16-Sep-1984 01:16:01  
14-Sep-1984 13:07:23

VAK-11 Bliss-32 V4.0-742  
[R]NOFF.SRC]NDXXTN.BLI;1

Page 2  
(1)

```

: 58          0058 1  |         Changed name of POOL.REQ to DMDEFS.REQ in NDXXTN.
: 59          0059 1  |
: 60          0060 1  |  --
: 61          0061 1  |
: 62          0062 1  |
: 63          0063 1  |  TABLE OF CONTENTS:
: 64          0064 1  |
: 65          0065 1  |
: 66          0066 1  |  FORWARD ROUTINE
: 67          0067 1  |    ASGXTN : NOVALUE,
: 68          0068 1  |    XTNPAG;
: 69          0069 1  |
: 70          0070 1  |
: 71          0071 1  |  INCLUDE FILES:
: 72          0072 1  |
: 73          0073 1  |
: 74          0074 1  |  LIBRARY 'NXPORT:XPORT';
: 75          0075 1  |
: 76          0076 1  |  SWITCHES LIST (REQUIRE);
: 77          0077 1  |
: 78          0078 1  |  REQUIRE 'REQ:PAGEN';
```

NDX]  
V04-

.....

R0079 1  
R0080 1  
R0081 1  
R0082 1  
R0083 1  
R0084 1  
R0085 1  
R0086 1  
R0087 1  
R0088 1  
R0089 1  
R0090 1  
R0091 1  
R0092 1  
R0093 1  
R0094 1  
R0095 1  
R0096 1  
R0097 1  
R0098 1  
R0099 1  
R0100 1  
R0101 1  
R0102 1  
R0103 1  
R0104 1  
R0105 1  
R0106 1  
R0107 1  
R0108 1  
R0109 1  
R0110 1  
R0111 1  
R0112 1  
R0113 1  
R0114 1  
R0115 1  
R0116 1  
R0117 1  
R0118 1  
R0119 1  
R0120 1  
R0121 1  
R0122 1  
R0123 1  
R0124 1  
R0125 1  
R0126 1  
R0127 1  
R0128 1  
R0129 1  
R0130 1  
R0131 1  
R0132 1  
R0133 1  
R0134 1  
R0135 1

Version: 'V04-000'

```
*****
*
* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
* ALL RIGHTS RESERVED.
*
* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
* TRANSFERRED.
*
* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
* CORPORATION.
*
* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
*
*****
```

++  
FACILITY: DSR (Digital Standard RUNOFF) / DSRPLUS

ABSTRACT:  
A page number carries with it not only its current value, but also  
codes as to how those values are to be displayed when they are finally  
output. It was decided to do it this way rather than have a separate  
table so that the program TCX would have less trouble.

ENVIRONMENT: Transportable BLISS

AUTHOR: Rich Friday

CREATION DATE: 1978

MODIFIED BY:

004 KAD00004 Keith Dawson 07-Mar-1983  
Global edit of all modules. Updated module names, idents,  
copyright dates. Changed require files to BLISS library.

--  
LITERAL  
page\_sct\_size = 4;

LITERAL  
sct\_chapt = 1; !Type of section:  
sct\_index = 2; ! Chapter section.  
sct\_append = 3; ! Index section.  
! Appendix section.

R  
R  
R  
R  
R  
R  
R  
R  
R  
R  
R









```
: R0218 1 !  
: R0219 1 LITERAL Offsets into pool control area (POOL) and pool area descriptor (PAD).  
: R0220 1 POOL_MAX_PADS = 0, !Maximum number of PADS that can be accommodated.  
: R0221 1 POOL_ACT_PADS = 1, !Current number of allocated PADS.  
: R0222 1 POOL_ACT_SIZE = 2; !Number of BPVALS in pool control area.  
: R0223 1  
: R0224 1 LITERAL  
: R0225 1 PAD_SIZE = 0, !Size of pooled area (BLISS VALUES).  
: R0226 1 PAD_ADDRESS = 1; !Start of pooled area.  
: R0227 1  
: R0228 1 ! The GET_SEG_ADDR macro returns the starting address of a segment from the  
: R0229 1 ! specified pool.  
: R0230 1 MACRO  
: MR0231 1 GET_SEG_ADDR(AREA,INDEX) =  
: MR0232 1 -BEGIN  
: MR0233 1 LOCAL  
: MR0234 1 PADTAB : REF VECTOR;  
: MR0235 1 PADTAB = .AREA+POOL_CNTRL_SIZE*%UPVAL;  
: MR0236 1 .PADTAB[PAD_CNTRL_SIZE*(INDEX-1)+PAD_ADDRESS]  
: MR0237 1 END  
: R0238 1 X;  
: R0239 1  
: R0240 1 !  
End of DMDEFS.REQ
```

NDXXTN  
V04-000

: 81  
: 82

0241 1  
0242 1 REQUIRE 'REQ:XTNTAB';

G 2  
16-Sep-1984 01:16:01  
14-Sep-1984 13:07:23

VAX-11 Bliss-32 V4.0-742  
[RUNOFF.SRC]NDXXTN.BLI;1

Page 8  
(1)

NDXX  
V04-

.....

R0243 1  
R0244 1  
R0245 1  
R0246 1  
R0247 1  
R0248 1  
R0249 1  
R0250 1  
R0251 1  
R0252 1  
R0253 1  
R0254 1  
R0255 1  
R0256 1  
R0257 1  
R0258 1  
R0259 1  
R0260 1  
R0261 1  
R0262 1  
R0263 1  
R0264 1  
R0265 1  
R0266 1  
R0267 1  
R0268 1  
R0269 1  
R0270 1  
R0271 1  
R0272 1  
R0273 1  
R0274 1  
R0275 1  
R0276 1  
R0277 1  
R0278 1  
R0279 1  
R0280 1  
R0281 1  
R0282 1  
R0283 1  
R0284 1  
R0285 1  
R0286 1  
R0287 1  
R0288 1  
R0289 1  
R0290 1  
R0291 1  
R0292 1  
R0293 1  
R0294 1  
R0295 1  
R0296 1  
R0297 1  
R0298 1  
R0299 1

Version: 'V04-000'

```

*****
*
* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
* ALL RIGHTS RESERVED.
*
* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
* TRANSFERRED.
*
* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
* CORPORATION.
*
* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
*
*****

```

++  
FACILITY: DSR (Digital Standard RUNOFF) / DSRPLUS

ABSTRACT:  
Parallel tables for associating index entries and pages.  
  
NOTE: The tables contain one extra entry, which is unused.  
That is so subtraction of 1 can be forgotten about.

ENVIRONMENT: Transportable BLISS

AUTHOR: Rich Friday

CREATION DATE: 1978

MODIFIED BY:  
  
002 KAD00002 Keith Dawson 07-Mar-1983  
Global edit of all modules. Updated module names, idents,  
copyright dates. Changed require files to BLISS library.

--  
LITERAL  
max\_xtn\_count = 100, !Maximum number of transaction numbers (condensed).  
  
!Number of BLISS values in a set of pages.  
  
xtn\_pagtab\_size = (max\_xtn\_count + 1) \* page\_sct\_size.

NDXXTN  
V04-000

1 2  
16-Sep-1984 01:16:01  
15-Sep-1984 22:54:49

VAX-11 Bliss-32 V4.0-742  
\_S255SDUA28:[RUNOFF.SRC]XTNTAB.REQ;1 Page 10  
(1)

NDXX  
V04-

```
: R0300 1 !Number of BLISS values in a list of transaction numbers.  
: R0301 1 !  
: R0302 1 ! xtn_xtntab_size = max_xtn_count + 1;  
: R0303 1 !  
: R0304 1 MACRO  
: R0305 1 ! xtn_define = VECTOR [xtn_xtntab_size] %,  
: R0306 1 ! xpagen_define = BLOCKVECTOR [max_xtn_ccunt + 1, page_sct_size] %;  
: R0307 1 !  
: R0308 1 ! End of XTNTAB.REQ
```

NDXXTN  
V04-000

J 2  
16-Sep-1984 01:16:01  
14-Sep-1984 13:07:23

VAX-11 Bliss-32 V4.0-742  
[RUNOFF.SRC]NDXXTN.BLI;1

Page 11  
(1)

NDXX  
V04-

```
: 83  
: 84  
: 85  
: 86  
: 87  
L 0309 1  
  0310 1 %IF %BLISS (BLISS32)  
  0311 1 %THEN  
  0312 1  
  0313 1 REQUIRE 'REQ:NDXVMSREQ';
```

: Rc

: 2

R0314 1  
R0315 1  
R0316 1  
R0317 1  
R0318 1  
R0319 1  
R0320 1  
R0321 1  
R0322 1  
R0323 1  
R0324 1  
R0325 1  
R0326 1  
R0327 1  
R0328 1  
R0329 1  
R0330 1  
R0331 1  
R0332 1  
R0333 1  
R0334 1  
R0335 1  
R0336 1  
R0337 1  
R0338 1  
R0339 1  
R0340 1  
R0341 1  
R0342 1  
R0343 1  
R0344 1  
R0345 1  
R0346 1  
R0347 1  
R0348 1  
R0349 1  
R0350 1  
R0351 1  
R0352 1  
R0353 1  
R0354 1  
R0355 1  
R0356 1  
R0357 1  
R0358 1  
R0359 1  
R0360 1  
R0361 1  
R0362 1  
R0363 1  
R0364 1  
R0365 1  
R0366 1  
R0367 1  
R0368 1  
R0369 1  
R0370 1

Version: 'V04-000'

```

*****
*
* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
* ALL RIGHTS RESERVED.
*
* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
* TRANSFERRED.
*
* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
* CORPORATION.
*
* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
*
*****

```

```

**
FACILITY:
  DSR (Digital Standard RUNOFF) /DSRPLUS DSRINDEX/INDEX Utility

```

```

ABSTRACT:
  This file contains external references to the error message numbers
  for DSRINDEX/INDEX.

  New messages must be defined in NDXVMSMSG.MSG and referenced here:
  both in the iACRO section (for DSRINDEX) and the EXTERNAL LITERAL
  section (for INDEX)

```

ENVIRONMENT: VAX/VMS User Mode

AUTHOR: JPK

CREATION DATE: 01-Feb-1983

```

MODIFIED BY:
  004      JPK00022      30-Mar-1983
           Modified NDXVMS, NDXFMT, NDXPAG, NDXVMSMSG and NDXVMSREQ
           to generate TEX output. Added module NDXTX.

  003      JPK00021      28-Mar-1983
           Modified NDXT20 to include E2.0 functionality.
           Modified NDXCLIDMP, NDXFMT, NDXPAG, NDXVRS to require RNODEF
           for BLISS36 and to remove any conditional require based on
           DSRPLUS_DEF.

```

NDXX  
V04-  
.....

NDXXTN  
V04--000

L 2  
16-Sep-1984 01:16:01  
15-Sep-1984 22:53:32

VAX-.1 Bliss-32 V4.0-742  
[RUNOFF.SRC]NDXVMSREQ.R32;1

Page 13  
(1)

NDXX  
V04--

```

: R0371 1 | 002 JPK00010 04-Feb-1983
: R0372 1 | | Cleaned up module names, modified revision history to
: R0373 1 | | conform with established standards. Updated copyright dates.
: R0374 1 | |
: R0375 1 | |
: R0376 1 | |
: R0377 1 | REQUIRE 'REQ:RNODEF';
```

.....

R0378 1  
R0379 1  
R0380 1  
R0381 1  
R0382 1  
R0383 1  
R0384 1  
R0385 1  
R0386 1  
R0387 1  
R0388 1  
R0389 1  
R0390 1  
R0391 1  
R0392 1  
R0393 1  
R0394 1  
R0395 1  
R0396 1  
R0397 1  
R0398 1  
R0399 1  
R0400 1  
R0401 1  
R0402 1  
R0403 1  
R0404 1  
R0405 1  
R0406 1  
R0407 1  
R0408 1  
R0409 1  
R0410 1  
R0411 1  
R0412 1  
R0413 1  
R0414 1  
R0415 1  
R0416 1  
R0417 1  
R0418 1  
R0419 1  
R0420 1  
R0421 1  
R0422 1  
R0423 1  
R0424 1  
R0425 1  
R0426 1  
R0427 1  
R0428 1  
R0429 1  
R0430 1  
R0431 1  
R0432 1  
R0433 1  
R0434 1

Version: 'V04-000'

\*\*\*\*\*  
\*  
\* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY \*  
\* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS. \*  
\* ALL RIGHTS RESERVED. \*  
\*  
\* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED \*  
\* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE \*  
\* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER \*  
\* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY \*  
\* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY \*  
\* TRANSFERRED. \*  
\*  
\* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE \*  
\* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT \*  
\* CORPORATION. \*  
\*  
\* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS \*  
\* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL. \*  
\*  
\*\*\*\*\*

++  
FACILITY: DSR (Digital Standard RUNOFF) / DSRPLUS  
ABSTRACT:  
Converts BLISS/VARIANT values into useful names.  
ENVIRONMENT: Transportable BLISS  
AUTHOR: Rich Friday  
CREATION DATE: 1978  
MODIFIED BY:  
016 KAD00016 Ray Marshall 19-Mar-1984  
Added GERMAN, FRENCH, & ITALIAN.  
015 KAD00015 Keith Dawson 18-Apr-1983  
Made the LN01 conditional the default for vanilla DSR --  
its value is 0 (no variant supplied).  
014 KAD00014 Keith Dawson 22-Mar-1983  
Asserted the LN01 conditional when DSRPLUS is asserted.  
013 KAD00013 Keith Dawson 20-Mar-1983  
Removed all references to .BIX and .BTC files.  
012 KAD00012 Keith Dawson 07-Mar-1983  
Global edit of all modules. Updated module names, idents,  
copyright dates. Changed require files to BLISS library.

Re  
L  
S  
Si  
Ru  
EL  
Li



R0435 1  
R0436 1  
R0437 1  
R0438 1  
R0439 1  
R0440 1  
R0441 1  
R0442 1  
R0443 1  
R0444 1  
R0445 1  
R0446 1  
R0447 1  
R0448 1  
R0449 1  
R0450 1  
R0451 1  
R0452 1  
R0453 1  
R0454 1  
R0455 1  
R0456 1  
R0457 1  
R0458 1  
R0459 1  
R0460 1  
R0461 1  
R0462 1  
R0463 1  
R0464 1  
R0465 1  
R0466 1  
R0467 1  
R0468 1  
R0469 1  
R0470 1  
R0471 2  
R0472 1  
R0473 1  
R0474 1  
R0475 1  
R0476 1  
R0477 1  
R0478 1  
R0479 1  
R0480 1  
R0481 2  
R0482 1  
R0483 1  
R0484 1  
R0485 1  
R0486 1  
R0487 1  
R0488 1  
R0489 1  
R0490 1  
R0491 1

```
--
++
DEFINITION OF /VARIANT BITS
The bit assignments are as follows:
Bit Weight Meaning
-----
--      0      If no /VARIANT is supplied (as for vanilla DSR),
              compile with LN01 support. LN01 support is also
              implied by the DSRPLUS variant.
      0      1      CLEAR = Unassigned
              SET   = Unassigned
      1      2      CLEAR = Normal compile
              SET   = Compile for DSRPLUS
      4-6    16     CLEAR = English (American) version
              SET   = 16 = German (Austrian)
                    32 = French
                    48 = Italian
--

-----
This variable (LN01) controls whether or not to compile an LN01-flavored
DSR. It is asserted by default, and also whenever DSRPLUS is asserted.
Modules utilizing LN01 are:
      DOOPTS  NOUT
COMPILETIME
      ln01 =
      ( (%VARIANT EQL 0) OR %VARIANT/2 )
      ;
-----
This variable (DSRPLUS) controls compilation for the DSRPLUS program.
All modules utilize DSRPLUS.
COMPILETIME
      dsrplus =
      ( %VARIANT/2 )
      ;
-----
This variable (FLIP) controls compilation of FLIP features of DSRPLUS.
It assures that FLIP features are compiled only on VMS systems.
Modules utilizing FLIP are many and various.
COMPILETIME
      flip =
```

: R0492 2  
: R0493 1  
: R0494 1  
: R0495 1  
: R0496 1  
: R0497 1  
: R0498 1  
: R0499 1  
: R0500 1  
: R0501 1  
: R0502 1  
: R0503 1  
: R0504 1  
: R0505 1  
: R0506 1  
: R0507 1

( %VARIANT/2 AND %BLISS(BLISS32) )

```
-----  
4-6 16 CLEAR = English (American) version  
SET = 16 = German (Austrian)  
32 = French  
48 = Italian  
COMPILETIME  
German = ( %VARIANT/16 AND NOT %VARIANT/32 AND NOT %VARIANT/64 ) ;  
COMPILETIME  
French = ( NOT %VARIANT/16 AND %VARIANT/32 AND NOT %VARIANT/64 ) ;  
COMPILETIME  
Italian = ( %VARIANT/16 AND %VARIANT/32 AND NOT %VARIANT/64 ) ;  
-----  
End of RNODEF.REQ
```

```

: R0508 1
: L R0509 1 %IF NOT DSRPLUS
: R0510 1 %THEN
: R0511 1
: R0512 1 MACRO
: R0513 1 INDEX$_BADLOGIC = DSRINDEX$_BADLOGIC %
: R0514 1 INDEX$_BADVALUE = DSRINDEX$_BADVALUE %
: R0515 1 INDEX$_INSVIRMEM = DSRINDEX$_INSVIRMEM %
: R0516 1 INDEX$_LINELENG = DSRINDEX$_LINELENG %
: R0517 1 INDEX$_NOREF = DSRINDEX$_NOREF %
: R0518 1 INDEX$_OPENIN = DSRINDEX$_OPENIN %
: R0519 1 INDEX$_OPENOUT = DSRINDEX$_OPENOUT %
: R0520 1 INDEX$_TOOMANY = DSRINDEX$_TOOMANY %
: R0521 1 INDEX$_VALERR = DSRINDEX$_VALERR %
: R0522 1 INDEX$_CANTBAL = DSRINDEX$_CANTBAL %
: R0523 1 INDEX$_CLOSEQUOT = DSRINDEX$_CLOSEQUOT %
: R0524 1 INDEX$_CONFQUAL = DSRINDEX$_CONFQUAL %
: R0525 1 INDEX$_CTRLCHAR = DSRINDEX$_CTRLCHAR %
: R0526 1 INDEX$_DOESNTFIT = DSRINDEX$_DOESNTFIT %
: R0527 1 INDEX$_DUPBEGIN = DSRINDEX$_DUPBEGIN %
: R0528 1 INDEX$_EMPTYIN = DSRINDEX$_EMPTYIN %
: R0529 1 INDEX$_IGNORED = DSRINDEX$_IGNORED %
: R0530 1 INDEX$_INVINPUT = DSRINDEX$_INVINPUT %
: R0531 1 INDEX$_INVRECORD = DSRINDEX$_INVRECORD %
: R0532 1 INDEX$_LASTCONT = DSRINDEX$_LASTCONT %
: R0533 1 INDEX$_NOBEGIN = DSRINDEX$_NOBEGIN %
: R0534 1 INDEX$_NOEND = DSRINDEX$_NOEND %
: R0535 1 INDEX$_NOINDEX = DSRINDEX$_NOINDEX %
: R0536 1 INDEX$_NOLIST = DSRINDEX$_NOLIST %
: R0537 1 INDEX$_OVERSTRK = DSRINDEX$_OVERSTRK %
: R0538 1 INDEX$_SKIPPED = DSRINDEX$_SKIPPED %
: R0539 1 INDEX$_SYNTAX = DSRINDEX$_SYNTAX %
: R0540 1 INDEX$_TEXTFILE = DSRINDEX$_TEXTFILE %
: R0541 1 INDEX$_TOODEEP = DSRINDEX$_TOODEEP %
: R0542 1 INDEX$_TOOFEW = DSRINDEX$_TOOFEW %
: R0543 1 INDEX$_TRUNCATED = DSRINDEX$_TRUNCATED %
: R0544 1 INDEX$_COMPLETE = DSRINDEX$_COMPLETE %
: R0545 1 INDEX$_CREATED = DSRINDEX$_CREATED %
: R0546 1 INDEX$_IDENT = DSRINDEX$_IDENT %
: R0547 1 INDEX$_PROCFILE = DSRINDEX$_PROCFILE %
: R0548 1 INDEX$_TEXT = DSRINDEX$_TEXT %
: R0549 1 INDEX$_TEXTD = DSRINDEX$_TEXTD %
: R0550 1 INDEX$_TMS11 = DSRINDEX$_TMS11 %
: R0551 1
: R0552 1 %FI
: R0553 1
: R0554 1 EXTERNAL LITERAL
: R0555 1 INDEX$_BADLOGIC, ! <internal: logic error detected>
: R0556 1 INDEX$_BADVALUE, ! <'!AS' is an invalid keyword value>
: R0557 1 INDEX$_INSVIRMEM, ! <insufficient virtual memory>
: R0558 1 INDEX$_LINELENG, ! <maximum line length is 120>
: R0559 1 INDEX$_NOREF, ! <page reference not found>
: R0560 1 INDEX$_OPENIN, ! <error opening '!AS' for input>
: R0561 1 INDEX$_OPENOUT, ! <error opening '!AS' for output>
: R0562 1 INDEX$_TOOMANY, ! <too many values supplied>
: R0563 1 INDEX$_VALERR, ! <specified value is out of legal range>
: R0564 1 INDEX$_CANTBAL, ! <can't balance last page>

```

```
: R0565 1 INDEX$_CLOSEQUOT, <missing close quote>
: R0566 1 INDEX$_CONQUAL, <conflicting qualifiers>
: R0567 1 INDEX$_CTRLCHAR, <the following line contains control characters - ignored>
: R0568 1 INDEX$_DOESNTFIT, <'!AD' will not fit at the current indentation level>
: R0569 1 INDEX$_DUPBEGIN, <duplicate .XPLUS (BEGIN) - inserted as .XPLUS (>>
: R0570 1 INDEX$_EMPTYIN, <empty input file '!AS'>
: R0571 1 INDEX$_IGNORED, <'!AS' ignored>
: R0572 1 INDEX$_INVINPUT, <invalid input file format in file '!AS'>
: R0573 1 INDEX$_INVRECORD, <invalid record type in file '!AS'>
: R0574 1 INDEX$_LASTCONT, <can't generate continuation heading on last page>
: R0575 1 INDEX$_NOBEGIN, <.XPLUS (END) with no .XPLUS (BEGIN) - inserted as .XPLUS (>>
: R0576 1 INDEX$_NOEND, <.XPLUS (BEGIN) has no corresponding .XPLUS (END)>
: R0577 1 INDEX$_NOINDEX, <no index information in file '!AS'>
: R0578 1 INDEX$_NOLIST, <parameter list not allowed>
: R0579 1 INDEX$_OVERSTRK, <the following line contains an overstrike sequence>
: R0580 1 INDEX$_SKIPPED, <!UL reference!XS inside page range - ignored>
: R0581 1 INDEX$_SYNTAX, <error parsing '!AS'>
: R0582 1 INDEX$_TEXTFILE, <error processing line !UL of TEX character file '!AS'>
: R0583 1 INDEX$_TOODEEP, <maximum subindex depth exceeded>
: R0584 1 INDEX$_TOOFEW, <not enough values supplied>
: R0585 1 INDEX$_TRUNCATED, <string too long - truncated>
: R0586 1 INDEX$_COMPLETE, <processing complete '!AS'>
: R0587 1 INDEX$_CREATED, <'!AS' created>
: R0588 1 INDEX$_IDENT, <INDEX version !AD>
: R0589 1 INDEX$_PROCFILE, <processing file '!AS'>
: R0590 1 INDEX$_TEXT, <!AS>
: R0591 1 INDEX$_TEXTD, <entry text: '!AD'>
: R0592 1 INDEX$_TMS11, <output file full - continuing with file '!AS'>
: R0593 1
```

```
88 0594 1
89 0595 1 %FI
90 0596 1
91 0597 1 SWITCHES LIST (NOREQUIRE);
92 0598 1
93 0599 1
94 0600 1 : MACROS:
95 0601 1 :
96 0602 1 :
97 0603 1 : EQUATED SYMBOLS.
98 0604 1 :
99 0605 1 :
100 0606 1 LITERAL
101 0607 1 TRUE = 1,
102 0608 1 FALSE = 0,
103 0609 1 XTN_MAX_SEGS = 100;
104 0610 1
105 0611 1
106 0612 1 :
107 0613 1 : OWN STORAGE:
108 0614 1 :
109 0615 1 :
110 0616 1 : EXTERNAL REFERENCES:
111 0617 1 :
112 0618 1 :
113 0619 1 EXTERNAL
114 0620 1 XPAGEN : REF XPAGEN_DEFINE,
115 0621 1 XTNCNT,
116 0622 1 XTNLSP : REF PAGE_DEFINITION,
117 0623 1 XTNLSX : REF VECTOR [XTN_MAX_SEGS + 1],
118 0624 1 XTNPOL : REF POOL,
119 0625 1 XTNSGP : REF BLOCK,
120 0626 1 XTNTAB : REF XTNTAB_DEFINE;
121 0627 1
122 0628 1 EXTERNAL ROUTINE
123 0629 1 GPOOL,
124 0630 1 PAGEQL,
125 0631 1 XPOOL;
126 0632 1
```

```
!Maximum number of pieces into
!which the transaction number
!tables can be broken.
```

```

128 0633 1 GLOBAL ROUTINE ASGXTN (PAGE, TRANSACTION) : NOVALUE = !
129 0634 1
130 0635 1 !++
131 0636 1 FUNCTIONAL DESCRIPTION:
132 0637 1
133 0638 1     Associates the current page number with a transaction
134 0639 1     number range.
135 0640 1
136 0641 1 FORMAL PARAMETERS:
137 0642 1
138 0643 1     PAGE indicates which page number is to be attached to
139 0644 1     the index entry.
140 0645 1     TRANSACTION, if not zero, is the highest transaction number
141 0646 1     to be associated with the given PAGE.
142 0647 1
143 0648 1 IMPLICIT INPUTS:
144 0649 1
145 0650 1     NONE
146 0651 1
147 0652 1 IMPLICIT OUTPUTS:
148 0653 1
149 0654 1     Implicit in this routine is the compression of the list of
150 0655 1     transaction numbers for a single page. Note that before
151 0656 1     the document page number is copied, a check is made to
152 0657 1     see if the previous transaction number refers to something
153 0658 1     on the same page. If that is the case, then no copy of the
154 0659 1     page number is made.
155 0660 1     This fact is important for the operation of the MODULE
156 0661 1     XPRT, which prints the index entries later.
157 0662 1     If the compression is not made, that module assumes that
158 0663 1     there are distinct pages having the same number; subsequently,
159 0664 1     it won't merge page numbers with 'to' or '-' correctly.
160 0665 1
161 0666 1 ROUTINE VALUE:
162 0667 1 COMPLETION CODES:
163 0668 1
164 0669 1     NONE
165 0670 1
166 0671 1 SIDE EFFECTS:
167 0672 1
168 0673 1     NONE
169 0674 1
170 0675 1 --
171 0676 1
172 0677 2 BEGIN
173 0678 2
174 0679 2 MAP
175 0680 2     PAGE : REF PAGE_DEFINITION;
176 0681 2
177 0682 2 LOCAL
178 0683 2     MERGE;
179 0684 2
180 0685 2     !Is this trip necessary??
181 0686 2
182 0687 2     IF
183 0688 2     .TRANSACTION EQL 0
184 0689 2     THEN

```

```
185      0690      RETURN;
186      0691
187      0692      !The first time through this code, initialize the pool.
188      0693      IF
189      0694      .XTNPOL EQL 0                               !First time through?
190      0695      THEN
191      0696          BEGIN
192      0697              !First, allocate the pool itself.
193      0698              ! (Extra slot gets pointer to XTNLX segment.)
194      0699              GPOOL (XTNPOL, XTN_MAX_SEGS + 1);
195      0700              !Now, allocate space for XTNLX.
196      0701              ! (Extra slot avoids having to subtract 1 all the time).
197      0702              XTNLX = XPOOL (XTNPOL, XTN_MAX_SEGS + 1);
198      0703          END;
199      0704
200      0705      !At this point at least a pool exists for saving the
201      0706      !segment information. However, the current segment, wherein
202      0707      !the transaction numbers and associated pages reside,
203      0708      !may be full, or even not yet allocated.
204      0709
205      0710      !In preparation for merging, see if the current page number and
206      0711      !last referenced page number are the same.
207      0712      IF
208      0713          !
209      0714          !.XTNLSP EQL 0
210      0715      THEN
211      0716          !There is no last page.
212      0717          MERGE = FALSE
213      0718      ELSE
214      0719          !Compare the two page numbers, taking display characteristics into account.
215      0720          MERGE = PAGEQL (.XTNLSP, .PAGE, TRUE);
216      0721
217      0722      IF
218      0723          !
219      0724          !.MERGE
220      0725      THEN
221      0726          !The transaction numbers refer to the same page of the
222      0727          !document. Just record the new highest transaction number.
223      0728          BEGIN
224      0729              XTNTAB [.XTNCNT] = .TRANSACTION;             !Record transaction in table, permanently.
225      0730              XTNLX [.XTNPOL [POOL_ACT_PADS]] = .TRANSACTION; !Remember it for next time around.
226      0731          RETURN;
227      0732          END;
228      0733
229      0734      !The new transaction number does not refer to the last
230      0735      !page, so no merge was possible. Allocate a new segment
231      0736      !if the current segment is either full, or else doesn't
232      0737      !exist.
233      0738      IF .XTNSGP EQL 0                                     !Any segment at all yet?
234      0739          !OR (.XTNCNT GEQ MAX_XTN_COUNT)               !Current segment full?
235      0740      THEN
236      0741          BEGIN
237      0742              !Allocate a new segment.
238      0743              !Note that the transaction numbers and page numbers
239      0744              !are saved in the same segment.
240      0745              XTNSGP = XPOOL (XTNPOL, XTN_XTNTAB_SIZE + XTN_PAGTAB_SIZE);
241      0746              !Make sure that a segment could be allocated.
```

```

: 00
: 1
: 1
: 1
: 9
:
:
:
:
: S
: R
: E
: L
```

```

242 0747 4 .XTNSGP EQL 0 OR (.XTNLSX EQL 0) !Catch no XTNLSX space here.
243 0748 3 THEN
244 0749 3 !The requested amount could not be allocated (pool full)
245 0750 4 BEGIN
246 L 0751 4 %IF %BLISS (BLISS32)
247 0752 4 %THEN ! Signal errors for BLISS32
248 0753 4
249 0754 4 SIGNAL_STOP (INDEX$_INSVIRMEM);
250 0755 4
251 U 0756 4 %ELSE ! Use $XPO_PUT_MSG otherwise
252 U 0757 4
253 U 0758 4 $XPO_PUT_MSG (SEVERITY = FATAL,
254 U 0759 4 STRING = 'can't extend transaction pool. ');
255 U 0760 4 %FI
256 0761 4
257 0762 4
258 0763 4 RETURN;
259 0764 3 END;
260 0765 3
261 0766 3 XTNCNT = 0; !No transaction numbers in this segment yet.
262 0767 3 XTNTAB = .XTNSGP; !Transaction table is at start of segment.
263 0768 3 !Page numbers are saved after transaction numbers.
264 0769 3 XPAGEN = .XTNSGP + XTN_XTNTAB_SIZE*%UPVAL;
265 0770 2 END;
266 0771 2
267 0772 2 !At this point, there is definitely a spot free to save the
268 0773 2 !transaction number and the associated page number.
269 0774 2 !That slot is the one AFTER the previous slot.
270 0775 2 XTNCNT = .XTNCNT + 1; !New transaction number slot.
271 0776 2 XTNTAB [0] = .XTNCNT; !Remember count in this list.
272 0777 2 XTNLSP = XPAGEN [.XTNCNT, SCT_TYP]; !Remember where this page is.
273 0778 3 BEGIN
274 0779 3 BIND !Ma! these structures
275 0780 3 COPY = XPAGEN [.XTNCNT, 0,0,0,0] : VECTOR; !vectors so that
276 0781 3 MAP !copying is easier.
277 0782 3 PAGE : REF VECTOR; !...
278 0783 3
279 0784 3 !Copy items one by one.
280 0785 3 INCR I FROM 0 TO (PAGE_SCT_SIZE -1) DO
281 0786 3 COPY [.I] = .PAGE [.I];
282 0787 2 END;
283 0788 2 !! XPAGEN [.XTNCNT, SCT_TYP] = .PAGE [SCT_TYP]; !Save this page number.
284 0789 2 !! XPAGEN [.XTNCNT, SCT_SUB_PAGE] = .PAGE [SCT SUB PAGE]; !...
285 0790 2 !! XPAGEN [.XTNCNT, SCT_NUMBER] = .PAGE [SCT NUMBER]; !...
286 0791 2 !! XPAGEN [.XTNCNT, SCT_PAGE] = .PAGE [SCT PAGE]; !...
287 0792 2 !! XPAGEN [.XTNCNT, SCT_DISPLAY] = .PAGE [SCT DISPLAY]; !...
288 0793 2 XTNTAB [.XTNCNT] = .TRANSACTION; !Record transaction in table, permanently.
289 0794 2 XTNLSX [.XTMPOL [POOL_ACT_PADS]] = .TRANSACTION; !Remember it for next time around.
290 0795 1 END; !End of ASGXTN

```

```

.TITLE NDXXTN
.IDENT \V04-000\

.EXTRN DSRINDEX$_BADLOGIC
.EXTRN DSRINDEX$_BADVALUE
.EXTRN DSRINDEX$_INSVIRMEM

```

: Le  
: Me  
: Co



```

.EXTRN DSRINDEX$_LINELENG
.EXTRN DSRINDEX$_NOREF
.EXTRN DSRINDEX$_OPENIN
.EXTRN DSRINDEX$_OPENOUT
.EXTRN DSRINDEX$_TOOMANY
.EXTRN DSRINDEX$_VALERR
.EXTRN DSRINDEX$_CANTBAL
.EXTRN DSRINDEX$_CLOSEQUOT
.EXTRN DSRINDEX$_CONFQUAL
.EXTRN DSRINDEX$_CTRLCHAR
.EXTRN DSRINDEX$_DOESNTFIT
.EXTRN DSRINDEX$_DUPBEGIN
.EXTRN DSRINDEX$_EMPTYIN
.EXTRN DSRINC_X$_IGNORED
.EXTRN DSRINDEX$_INVINPUT
.EXTRN DSRINDEX$_INVRECORD
.EXTRN DSRINDEX$_LASTCONT
.EXTRN DSRINDEX$_NOBEGIN
.EXTRN DSRINDEX$_NOEND
.EXTRN DSRINDEX$_NOINDEX
.EXTRN DSRINDEX$_NOLIST
.EXTRN DSRINDEX$_OVERSTRK
.EXTRN DSRINDEX$_SKIPPED
.EXTRN DSRINDEX$_SYNTAX
.EXTRN DSRINDEX$_TEXTFILE
.EXTRN DSRINDEX$_TOODEEP
.EXTRN DSRINDEX$_TOOFEW
.EXTRN DSRINDEX$_TRUNCATED
.EXTRN DSRINDEX$_COMPLETE
.EXTRN DSRINDEX$_CREATED
.EXTRN DSRINDEX$_IDENT
.EXTRN DSRINDEX$_PROCFILE
.EXTRN DSRINDEX$_TEXT, DSRINDEX$_TEXTD
.EXTRN DSRINDEX$_TMS11
.EXTRN XPAGEN, XTNCNT, XTNLSP
.EXTRN XTNLX, XTNPOL, XTNSGP
.EXTRN XTNTAB, GPOOL, PAGEQL
.EXTRN XPOOL

```

.PSECT \$CODE\$,NOWRT,2

```

OFFC 00000
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
55 00000000G EF 9E 0002C
54 00000000G EF 9E 00033
53      08 AC D0 0003A
      01 12 0003E
      65 D5 00041 1$:
      19 12 00043
7E      65 8F 9A 00045

```

```

.ENTRY ASGXTN, Save R2,R3,R4,R5,R6,R7,R8,R9,R10,- : 0633
R11
MOVAB XPAGEN, R11
MOVAB XTNLSP, R10
MOVAB XPOOL, R9
MOVAB XTNLX, R8
MOVAB XTNSGP, R7
MOVAB XTNTAB, R6
MOVAB XTNPOL, R5
MOVAB XTNCNT, R4
MOVL TRANSACTION, R3 : 0688
BNEQ 1$
RET
TSTL XTNPOL : 0694
BNEQ 2$
MOVZBL #101, -(SP) : 0699

```

00000000G	EF		55	DD	00049	PUSHL	R5		
	7E	65	02	FB	0004B	CALLS	#2, GPOOL		0702
			8F	9A	00052	MOVZBL	#101, -(SP)		
	69		55	DD	00056	PUSHL	R5		
	68		02	FB	00058	CALLS	#2, XPOOL		
	51		50	D0	0005B	MOVL	R0, XTNLX		
			6A	D0	0005E	2\$:	MOVL	XTNLSP, R1	0713
			04	12	00061	BNEQ	3\$		
			50	D4	00063	CLRL	MERGE		0716
			0E	11	00065	BRB	4\$		
			01	DD	00067	3\$:	PUSHL	#1	0719
		04	AC	DD	00069	PUSHL	PAGE		
			51	DD	0006C	PUSHL	R1		
00000000G	EF		03	FB	0006E	CALLS	#3, PAGEQL		
	0A		50	E9	00075	4\$:	BLBC	MERGE, 5\$	0722
	50		64	D0	00078	MOVL	XTNCNT, R0		0727
00 B640			53	D0	0007B	MOVL	R3, @XTNTAB[R0]		
			5F	11	00080	BRB	11\$		0728
			67	D5	00082	5\$:	TSTL	XTNSGP	0736
			09	13	00084	BEQL	6\$		
00000064	8F		64	D1	00086	CMP	XTNCNT, #100		0737
			2E	19	0008D	BLSS	9\$		
	7E	01F9	8F	3C	0008F	6\$:	MOVZWL	#505, -(SP)	0743
			55	DD	00094	PUSHL	R5		
	69		02	FB	00096	CALLS	#2, XPOOL		
	67		50	D0	00099	MOVL	R0, XTNSGP		
			04	13	0009C	BEQL	7\$		0747
			68	D5	0009E	TSTL	XTNLX		
			0E	12	000A0	BNEQ	8\$		
		00000000G	8F	DD	000A2	7\$:	PUSHL	#DSRINDEX\$, INSVIRMEM	0754
00000000G	00		01	FB	000A8	CALLS	#1, LIB\$STOP		
			04	000AF	RET				0750
			64	D4	000B0	8\$:	CLRL	XTNCNT	0766
	66		67	D0	000B2	MOVL	XTNSGP, XTNTAB		0767
6B	67	00000194	8F	C1	000B5	ADDL3	#404, XTNSGP, XPAGEN		0769
			64	D6	000BD	9\$:	INCL	XTNCNT	0775
	51		64	D0	000BF	MOVL	XTNCNT, R1		0776
	00 B6		51	D0	000C2	MOVL	R1, @XTNTAB		
52	51		04	78	000C6	ASHL	#4, R1, R2		0777
	52		6B	C0	000CA	ADDL2	XPAGEN, R2		
	6A		52	D0	000CD	MOVL	R2, XTNLSP		
			50	D4	000D0	CLRL	I		0786
	6240	04 BC40	00	D0	000D2	10\$:	MOVL	@PAGE[I], (R2)[I]	
F6	50		03	F3	000D8	AOBLEQ	#3, I, 10\$		
	00 B641		53	D0	000DC	MOVL	R3, @XTNTAB[R1]		0793
	50		65	D0	000E1	11\$:	MOVL	XTNPOL, R0	0794
	50		04	C0	000E4	ADDL2	#4, R0		
	50		60	D0	000E7	MOVL	(R0), R0		
	00 B840		53	D0	000EA	MOVL	R3, @XTNLX[R0]		
			04	000EF	RET				0795

; Routine Size: 240 bytes. Routine Base: \$CODE\$ + 0000

; 291 0796 1

```

: 293 0797 1 GLOBAL ROUTINE XTNPAG (TRANSACTION) = !
: 294 0798 1
: 295 0799 1 ++
: 296 0800 1 FUNCTIONAL DESCRIPTION:
: 297 0801 1
: 298 0802 1     Given a transaction number, return the address of
: 299 0803 1     the corresponding page number.
: 300 0804 1
: 301 0805 1 FORMAL PARAMETERS:
: 302 0806 1
: 303 0807 1     TRANSACTION - The transaction number.
: 304 0808 1
: 305 0809 1 IMPLICIT INPUTS:
: 306 0810 1
: 307 0811 1     NONE
: 308 0812 1
: 309 0813 1 IMPLICIT OUTPUTS:
: 310 0814 1
: 311 0815 1     NONE
: 312 0816 1
: 313 0817 1 ROUTINE VALUE:
: 314 0818 1 COMPLETION CODES:
: 315 0819 1
: 316 0820 1     Address of the corresponding page number.
: 317 0821 1
: 318 0822 1 SIDE EFFECTS:
: 319 0823 1
: 320 0824 1     NONE
: 321 0825 1
: 322 0826 1 --
: 323 0827 1
: 324 0828 2 BEGIN
: 325 0829 2
: 326 0830 2 IF !
: 327 0831 2     .TRANSACTION EQL 0
: 328 0832 2 THEN
: 329 0833 2     RETURN 0;
: 330 0834 2
: 331 0835 2 !Find the correct segment number.
: 332 0836 2 !NOTE: Start at 2 because first is XTNLX.
: 333 0837 2
: 334 0838 2 INCR I FROM 2 TO .XTNPOL [POOL_ACT_PADS] DO
: 335 0839 3 BEGIN
: 336 0840 3
: 337 0841 3 IF !
: 338 0842 3     .TRANSACTION LEQ .XTNLX [.I]
: 339 0843 3 THEN
: 340 0844 3 !Search segment for exact transaction number.
: 341 0845 3 !That results in an index into the corresponding
: 342 0846 3 !set of saved pages.
: 343 0847 4 BEGIN
: 344 0848 4
: 345 0849 4 LOCAL
: 346 0850 4     XTN_TABLE : REF XTNTAB DEFINE,
: 347 0851 4     XPAGEN : REF XPAGEN_DEFINE;
: 348 0852 4
: 349 0853 4     XTN_TABLE = GET_SEG_ADDR (XTNPOL, .I);

```

```

350 0854 4      XPAGEN = GET_SEG_ADDR (XTNPOL, .I) + XTN_XTN_TAB_SIZE*%UPVAL;
351 0855 4
352 0856 4      INCR J FROM 1 TO .XTN_TABLE [0] DO
353 0857 5      BEGIN
354 0858 5
355 0859 5      IF
356 0860 5          .TRANSACTION LEQ .XTN_TABLE [ J ]
357 0861 5      THEN
358 0862 5          RETURN XPAGEN [J, SCT_TYP]
359 0863 5
360 0864 4      END;
361 0865 4
362 0866 3      END;
363 0867 3
364 0868 2      END;
365 0869 2
366 L 0870 2  %IF %BLISS (BLISS32)
367 0871 2  %THEN
368 0872 2          ! Signal errors for BLISS32
369 0873 2  SIGNAL_STOP (INDEX$_NOREF, 0, INDEX$_BADLOGIC);
370 0874 2
371 U 0875 2  %ELSE
372 U 0876 2          ! Use $XPO_PUT_MSG otherwise
373 U 0877 2  $XPO_PUT_MSG (SEVERITY = FATAL,
374 U 0878 2  STRING = 'internal error - page reference not found. ');
375 U 0879 2
376 0880 2  %FI
377 0881 2
378 0882 2  RETURN (
379 0883 2
380 0884 2  EXTERNAL
381 0885 2  PAGEN;
382 0886 2
383 0887 2  PAGEN)
384 0888 1  END;

```

.EXTRN PAGEN

```

          007C 00000
          56      04  AC  D0 00002
          68      13 00006
          54 00000000G EF  D0 00008
          51      01  D0 0000F
          3A      11 00012
          00000000GFF41 56  D1 00014 1$:
          30      14 0001C
          53      0C  A4  9E 00C1E
          52      53  D0 00022
          50      51  01  78 0C025
          55      FC  A240 D0 00029
          52      FC A340 00000194 8F  C1 0002E
          53      D4 00038
          6543      0E  11 0003A
          56      D1 0003C 2$:
          08      14 00040
          .ENTRY XTNPAG, Save R2,R3,R4,R5,R6
          MOVL TRANSACTION, R6
          BEQL 5$
          MOVL XTNPOL, R4
          MOVL #1, I
          BRB 4$
          CMPL P6, @XTNLSA[I]
          BGTR 4$
          MOVAB 12(R4), R3
          MOVL R3, PADTAB
          ASHL #1, I, R0
          MOVL -4(PADTAB)[R0], XTN_TABLE
          ADDL3 #404, -4(PADTAB)[R0], XPAGEN
          CLRL J
          BRB 3$
          CMPL R6, (XTN_TABLE)[J]
          BGTR 3$
          : 0797
          : 0831
          : 0838
          : 0842
          : 0853
          : 0854
          : 0856
          : 0860

```

```

50          53          04 78 00042      ASHL    #4, J, R0
50          50          52 C0 00046      ADDL2   XPAGEN, R0
          EE          04 04 00049      RET
          C1          53          65 F3 0004A 3$:  AOBLEQ  (XTN TABLE), J, 2$
          51          51          A4 F3 0004E 4$:  AOBLEQ  4(R4), I, 1$
          00000000G    04          8F DC 00053      PUSHL  #DSRINDEX$_BADLOGIC
          00000000G    7E D4 00059      CLRL   -(SP)
          00000000G    8F DD 0005B      PUSHL  #DSRINDEX$_NOREF
          00000000G    03 FB 00061      CALLS  #3, LIB$STOP
          50 00000000G    EF 9E 00068      MOVAB  PAGEN, R0
          04 0006F      RET
          50 D4 00070 5$:  CLRL   R0
          04 00072      RET
    
```

```

: 0862
: 0857
: 0838
: 0873
:
: 0882
: 0888
    
```

: Routine Size: 115 bytes, Routine Base: \$CODE\$ + 00F0

```

: 385      0889 1
: 386      0890 1 END
: 387      0891 1
: 388      0892 0 ELUDOM
    
```

!End of module

.EXTRN LIB\$STOP

PSECT SUMMARY

Name	Bytes	Attributes
\$CODE\$	355	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

COMMAND QUALIFIERS

BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS\$:NDXXTN/OBJ=OBJ\$:NDXXTN MSRC\$:NDXXTN/UPDATE=(ENHS:NDXXTN)

```

: Size:      355 code + 0 data bytes
: Run Time:   00:11.1
: Elapsed Time: 00:25.2
: Lines/CPU Min: 4830
    
```

NDXXTN  
V04-000

N 3  
16-Sep-1984 01:16:01

VAX-11 Bliss-32 V4.0-742

Page 28

: Lexemes/CPU-Min: 12043  
: Memory Used: 88 pages  
: Compilation Complete

NM  
V04-

.....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			