K 7

```
RRRRRRRRRRRR     UUU         UUU  NNN         NNN     000000000     FFFFFFFFFFFFFFF  FFFFFFFFFFFFFFF
RRRRRRRRRRRR     UUU         UUU  NNN         NNN     000000000     FFFFFFFFFFFFFFF  FFFFFFFFFFFFFFF
RRRRRRRRRRRR     UUU         UUU  NNN         NNN     000000000     FFFFFFFFFFFFFFF  FFFFFFFFFFFFFFF
RRR         RRR  UUU         UUU  NNN         NNN  000         000  FFF              FFF
RRR         RRR  UUU         UUU  NNN         NNN  000         000  FFF              FFF
RRR         RRR  UUU         UUU  NNN         NNN  000         000  FFF              FFF
RRR         RRR  UUU         UUU  NNNNNN      NNN  000         000  FFF              FFF
RRR         RRR  UUU         UUU  NNNNNN      NNN  000         000  FFF              FFF
RRR         RRR  UUU         UUU  NNNNNN      NNN  000         000  FFF              FFF
RRRRRRRRRRRR     UUU         UUU  NNN   NNN   NNN  000         000  FFFFFFFFFFFF     FFFFFFFFFFFF
RRRRRRRRRRRR     UUU         UUU  NNN   NNN   NNN  000         000  FFFFFFFFFFFF     FFFFFFFFFFFF
RRRRRRRRRRRR     UUU         UUU  NNN   NNN   NNN  000         000  FFFFFFFFFFFF     FFFFFFFFFFFF
RRR   RRR        UUU         UUU  NNN      NNNNNN  000         000  FFF              FFF
RRR   RRR        UUU         UUU  NNN      NNNNNN  000         000  FFF              FFF
RRR   RRR        UUU         UUU  NNN      NNNNNN  000         000  FFF              FFF
RRR      RRR     UUU         UUU  NNN         NNN  000         000  FFF              FFF
RRR      RRR     UUU         UUU  NNN         NNN  000         000  FFF              FFF
RRR      RRR     UUU         UUU  NNN         NNN  000         000  FFF              FFF
RRR         RRR  UUUUUUUUUUUUUUU  NNN         NNN     000000000     FFF              FFF
RRR         RRR  UUUUUUUUUUUUUUU  NNN         NNN     000000000     FFF              FFF
RRR         RRR  UUUUUUUUUUUUUUU  NNN         NNN     000000000     FFF              FFF
```

```
_$2

Sym
---
NDX
NDX
NUM
NUM
OPE
OUT
PAC
PAC
PAC
PAC
PAC
PAC
PAC
PAD
PAG
PAG
PAG
PAG
PAG
PAG
PAG
PER
PUT
RCO
RIN

RLI
RNO
RNO
RTY
SAV
STR
STR
STR
STR
STR
STR

STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
```

FILEID**CONVBB

```
  CCCCCCCC    OOOOOO    NN      NN  VV      VV  BBBBBBBB    BBBBBBBB
  CCCCCCCC    OOOOOO    NN      NN  VV      VV  BBBBBBBB    BBBBBBBB
CC          OO      OO  NN      NN  VV      VV  BB      BB  BB      BB
CC          OO      OO  NN      NN  VV      VV  BB      BB  BB      BB
CC          OO      OO  NNNN    NN  VV      VV  BB      BB  BB      BB
CC          OO      OO  NNNN    NN  VV      VV  BB      BB  BB      BB
CC          OO      OO  NN  NN  NN  VV      VV  BBBBBBBB    BBBBBBBB
CC          OO      OO  NN  NN  NN  VV      VV  BBBBBBBB    BBBBBBBB
CC          OO      OO  NN    NNNN  VV      VV  BB      BB  BB      BB
CC          OO      OO  NN    NNNN  VV      VV  BB      BB  BB      BB
CC          OO      OO  NN      NN    VV  VV    BB      BB  BB      BB      ....
CC          OO      OO  NN      NN    VV  VV    BB      BB  BB      BB      ....
  CCCCCCCC    OOOOOO    NN      NN      VV      BBBBBBBB    BBBBBBBB        ....
  CCCCCCCC    OOOOOO    NN      NN      VV      BBBBBBBB    BBBBBBBB        ....

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

```
  0 MODULE CONVBB (                                   !
  0                     IDENT = 'V04-000'
P 0 %BLISS32[,
P 0         ADDRESSING_MODE(EXTERNAL=LONG_RELATIVE,NONEXTERNAL=LONG_RELATIVE)
  0         ]
  0                     ) =
  1 BEGIN
  1 !
  1 !****************************************************************************
  1 !*                                                                          *
  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
  1 !*  ALL RIGHTS RESERVED.                                                    *
  1 !*                                                                          *
  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
  1 !*  TRANSFERRED.                                                            *
  1 !*                                                                          *
  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
  1 !*  CORPORATION.                                                            *
  1 !*                                                                          *
  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
  1 !*                                                                          *
  1 !*                                                                          *
  1 !****************************************************************************
  1 !
  1 !++
  1 ! FACILITY:      DSR (Digital Standard RUNOFF) / DSRPLUS
  1 !
  1 ! ABSTRACT: Convert a binary number into a vector of characters and
  1 !           return the result and character count.
  1 !
  1 !
  1 !
  1 ! ENVIRONMENT: Transportable
  1 !
  1 ! AUTHOR: R.W.Friday     CREATION DATE: May, 1979
  1 !
```

```
%SBTTL 'Revision History'
!
! MODIFIED BY:
!
!         002     KFA00002        Ken Alden       07-Mar-1983
!                 Global edit of all modules. Updated module names, idents,
!                 copyright dates. Changed require files to BLISS library.
!
!--
```

```
: 54         0052   1 %SBTTL 'Module Level Declarations'
: 55         0053   1
: 56         0054   1 !
: 57         0055   1
```

```
1 GLOBAL ROUTINE CONVBB (BINARY_NUMBER, KHARACTERS, KHARACTER_COUNT, BASE) : NOVALUE =
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       Converts 'binary_number' to a vector of characters,
1 !       returning them in 'kharacters'; kharacter_count is the
1 !       number of digits converted.
1 !       The absolute value of 'binary_number' is converted,
1 !       so that the user is responsible for handling negative numbers.
1 !       The number will be converted according to the value of BASE.
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !       See FUNCTIONAL DESCRIPTION
1 !
1 ! IMPLICIT INPUTS:
1 !
1 !       NONE
1 !
1 ! IMPLICIT OUTPUTS:
1 !
1 !       NONE
1 !
1 ! ROUTINE VALUE:
1 ! COMPLETION CODES:
1 !
1 !       NONE
1 !
1 ! SIDE EFFECTS:
1 !
1 !       NONE
1 !
1 !--
1
2     BEGIN
2
2     OWN
2         DIGITS : INITIAL (CH$PTR(UPLIT('0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ')));
2
2     MAP
2         KHARACTERS : REF VECTOR;
2
2     LOCAL
2         LEFT_TO_CONVERT;
2
2     .KHARACTER_COUNT = 0;
2     LEFT_TO_CONVERT = ABS (.BINARY_NUMBER);
2
2     DO
3         BEGIN
3         KHARACTERS [..KHARACTER_COUNT] = CH$RCHAR( CH$PLUS(.DIGITS, (.LEFT_TO_CONVERT MOD .BASE)));
3         LEFT_TO_CONVERT = .LEFT_TO_CONVERT/.BASE;
3         .KHARACTER_COUNT = ..KHARACTER_COUNT + 1;
3         END
2     UNTIL .LEFT_TO_CONVERT EQL 0;
2
```

```
                                                                  K 15
CONVBB                                                            16-Sep-1984 00:10:31    VAX-11 Bliss-32 V4.0-742             Page  5
V04-000          Module Level Declarations                        14-Sep-1984 13:05:52    [RUNOFF.SRC]CONVBB.BLI;1                   (4)

;  116           0113  2     RETURN;
;  117           0114  1     END;                                        !End of CONVBB


                                                                        .TITLE  CONVBB
                                                                        .IDENT  \V04-000\

                                                                        .PSECT  $PLIT$,NOWRT,NOEXE,2

45  44  43  42  41  39  38  37  36  35  34  33  32  31  30  00000 P.AAA:  .ASCII  \0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ\          ;
54  53  52  51  50  4F  4E  4D  4C  4B  4A  49  48  47  46  0000F                                                                  ;
                                    5A  59  58  57  56  55  0001E                                                                  ;

                                                                        .PSECT  $OWN$,NOEXE,2

                                             00000000' 00000 DIGITS:  .ADDRESS P.AAA                                               ;


                                                                        .PSECT  $CODE$,NOWRT,2

                                                  0004 00000            .ENTRY  CONVBB, Save R2                                    ; 0056
                                        0C     BC   D4 00002            CLRL    @KHARACTER_COUNT                                   ; 0102
                            52          04     AC   D0 00005            MOVL    BINARY_NUMBER, LEFT_TO_CONVERT                     ; 0103
                                               03   18 00009            BGEQ    1$                                                 ;
                            52                 52   CE 0000B            MNEGL   LEFT_TO_CONVERT, LEFT_TO_CONVERT                   ;
                            51          0C     BC   D0 0000E 1$:        MOVL    @KHARACTER_COUNT, R1                               ; 0107
          7E           00   52                 01   7A 00012            EMUL    #1, LEFT_TO_CONVERT, #0, -(SP)                     ;
          50           50   8E          10     AC   7B 00017            EDIV    BASE, (SP)+, R0, R0                                ;
                       08 BC41   00000000'FF40      9A 0001D            MOVZBL  @DIGITS[R0], @KHARACTERS[R1]                       ;
                            52          10     AC   C6 00027            DIVL2   BASE, LEFT_TO_CONVERT                              ; 0108
                                        0C     BC   D6 0002B            INCL    @KHARACTER_COUNT                                   ; 0109
                                               52   D5 0002E            TSTL    LEFT_TO_CONVERT                                    ; 0111
                                               DC   12 00030            BNEQ    1$                                                 ;
                                                    04 00032            RET                                                        ; 0114

; Routine Size:  51 bytes,    Routine Base:  $CODE$ + 0000


;  118           0115  1
;  119           0116  1 END                                             !End of module
;  120           0117  0 ELUDOM




;                                   PSECT SUMMARY
;
;      Name                   Bytes                    Attributes
;
;   $PLIT$                       36  NOVEC,NOWRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;   $OWN$                         4  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;   $CODE$                       51  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;                                   COMMAND QUALIFIERS

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

; Size:           51 code + 40 data bytes
; Run Time:          00:01.7
; Elapsed Time:      00:05.7
; Lines/CPU Min:      4153
; Lexemes/CPU-Min:   9230
; Memory Used:  23 pages
; Compilation Complete
```

0338 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY