G 6

```
PPPPPPPPPPPP        AAAAAAAAA        SSSSSSSSSSSS  RRRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
PPPPPPPPPPPP        AAAAAAAAA        SSSSSSSSSSSS  RRRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
PPPPPPPPPPPP        AAAAAAAAA        SSSSSSSSSSSS  RRRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
PPP         PPP  AAA         AAA  SSS              RRR         RRR        TTT        LLL
PPP         PPP  AAA         AAA  SSS              RRR         RRR        TTT        LLL
PPP         PPP  AAA         AAA  SSS              RRR         RRR        TTT        LLL
PPP         PPP  AAA         AAA  SSS              RRR         RRR        TTT        LLL
PPP         PPP  AAA         AAA  SSS              RRR         RRR        TTT        LLL
PPP         PPP  AAA         AAA  SSS              RRR         RRR        TTT        LLL
PPPPPPPPPPPP     AAA         AAA     SSSSSSSSS     RRRRRRRRRRRR           TTT        LLL
PPPPPPPPPPPP     AAA         AAA     SSSSSSSSS     RRRRRRRRRRRR           TTT        LLL
PPPPPPPPPPPP     AAA         AAA     SSSSSSSSS     RRRRRRRRRRRR           TTT        LLL
PPP              AAAAAAAAAAAAAAA              SSS  RRR   RRR              TTT        LLL
PPP              AAAAAAAAAAAAAAA              SSS  RRR   RRR              TTT        LLL
PPP              AAAAAAAAAAAAAAA              SSS  RRR   RRR              TTT        LLL
PPP              AAA         AAA              SSS  RRR      RRR           TTT        LLL
PPP              AAA         AAA              SSS  RRR      RRR           TTT        LLL
PPP              AAA         AAA              SSS  RRR      RRR           TTT        LLL
PPP              AAA         AAA  SSSSSSSSSSSS     RRR         RRR        TTT        LLLLLLLLLLLLLLL
PPP              AAA         AAA  SSSSSSSSSSSS     RRR         RRR        TTT        LLLLLLLLLLLLLLL
PPP              AAA         AAA  SSSSSSSSSSSS     RRR         RRR        TTT        LLLLLLLLLLLLLLL
```

```
_$2

Sym
---

PAS
PAS
PAS
PAS
PAS
PAS
PAS
PAS
PAS
PAS
PAS
PAS
PAS
PAS
PAS
PAS
PAS
PAS
PAS
PAS
PAS
PAS
PAS
PAS
PAS
PAS
PAS
PAS

PAS

PAS

PAS

PAS
PAS
PAS
PAS
PAS

PAS
PAS
PAS
PAS
PAS
PAS
PAS
PAS
```

FILEID**PASWRIBIN

E 11

```
PPPPPPPP      AAAAAA      SSSSSSSS  WW      WW  RRRRRRRR      IIIIII    BBBBBBBB      IIIIII    NN      NN
PPPPPPPP      AAAAAA      SSSSSSSS  WW      WW  RRRRRRRR      IIIIII    BBBBBBBB      IIIIII    NN      NN
PP      PP  AA      AA  SS          WW      WW  RR      RR      II      BB      BB      II      NN      NN
PP      PP  AA      AA  SS          WW      WW  RR      RR      II      BB      BB      II      NN      NN
PP      PP  AA      AA  SS          WW      WW  RR      RR      II      BB      BB      II      NNNN    NN
PP      PP  AA      AA  SS          WW      WW  RR      RR      II      BB      BB      II      NNNN    NN
PPPPPPPP    AA      AA    SSSSSS    WW      WW  RRRRRRRR        II      BBBBBBBB        II      NN  NN  NN
PPPPPPPP    AA      AA    SSSSSS    WW      WW  RRRRRRRR        II      BBBBBBBB        II      NN  NN  NN
PP          AAAAAAAAAA          SS  WW  WW  WW  RR  RR          II      BB      BB      II      NN    NNNN
PP          AAAAAAAAAA          SS  WW  WW  WW  RR  RR          II      BB      BB      II      NN    NNNN
PP          AA      AA          SS  WWWW  WWWW  RR    RR        II      BB      BB      II      NN      NN      ....
PP          AA      AA          SS  WWWW  WWWW  RR    RR        II      BB      BB      II      NN      NN      ....
PP          AA      AA  SSSSSSSS    WW      WW  RR      RR    IIIIII    BBBBBBBB      IIIIII    NN      NN      ....
PP          AA      AA  SSSSSSSS    WW      WW  RR      RR    IIIIII    BBBBBBBB      IIIIII    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
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

```
0 MODULE PAS$WRITE_BIN ( %TITLE 'Write value in base 2'
0                   IDENT = '1-002'                  ! File: PASWRIBIN.B32 Edit: SBL1002
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 !++
1 ! FACILITY:     Pascal Language Support
1 !
1 ! ABSTRACT:
1 !
1 !       This module contains procedures which write a value in
1 !       base 2 to a textfile or string.
1 !
1 ! ENVIRONMENT:  User mode - AST reentrant
1 !
1 ! AUTHOR: Steven B. Lionel, CREATION DATE: 1-April-1981
1 !
1 ! MODIFIED BY:
1 !
1 ! 1-001 - Original.  SBL 1-April-1981
1 ! 1-002 - Make total-width a longword.  SBL 29-June-1982
1 !--
1
```

```
:    49        0048  1 %SBTTL 'Declarations'
:    50        0049  1 !
:    51        0050  1 ! PROLOGUE DEFINITIONS:
:    52        0051  1 !
:    53        0052  1
:    54        0053  1 REQUIRE 'RTLIN:PASPROLOG';                     ! Externals, linkages, PSECTs, structures
:    55        0117  1
:    56        0118  1 !
:    57        0119  1 ! TABLE OF CONTENTS:
:    58        0120  1 !
:    59        0121  1
:    60        0122  1 FORWARD ROUTINE
:    61        0123  1     PAS$WRITE_BIN: NOVALUE,                    ! Write to textfile
:    62        0124  1     PAS$WRITEV_BIN: NOVALUE;                   ! Write to string
:    63        0125  1
:    64        0126  1 !
:    65        0127  1 ! MACROS:
:    66        0128  1 !
:    67        0129  1 !          NONE
:    68        0130  1 !
:    69        0131  1 ! EQUATED SYMBOLS:
:    70        0132  1 !
:    71        0133  1 !          NONE
:    72        0134  1 !
:    73        0135  1 ! FIELDS:
:    74        0136  1 !
:    75        0137  1 !          NONE
:    76        0138  1 !
:    77        0139  1 ! OWN STORAGE:
:    78        0140  1 !
:    79        0141  1 !          NONE
:    80        0142  1 !
:    81        0143  1
:    82        0144  1 !! If this is for a V2 system, redefine OTS$CVT_L_TB as PAS$CVT_L_TB.
:    83      L 0145  1 %IF %VARIANT
:    84      U 0146  1 %THEN
:    85      U 0147  1 UNDECLARE
:    86      U 0148  1     OTS$CVT_L_TB;
:    87      U 0149  1 EXTERNAL ROUTINE
:    88      U 0150  1     PAS$CVT_L_TB;
:    89      U 0151  1 BIND ROUTINE
:    90      U 0152  1     OTS$CVT_L_TB = PAS$CVT_L_TB;
:    91        0153  1 %FI
```

```
;    93      0154  1 %SBTTL 'PAS$WRITE_BIN - Write a value in binary'
;    94      0155  1 GLOBAL ROUTINE PAS$WRITE_BIN (                          ! Write a value in binary
;    95      0156  1         PFV: REF $PAS$PFV_FILE_VARIABLE,                ! File variable
;    96      0157  1         NBITS,                                          ! Size of value in bits
;    97      0158  1         VALUE,                                          ! Address of value
;    98      0159  1         TOTAL_WIDTH: SIGNED,                            ! Total field width
;    99      0160  1         ERROR,                                          ! Error unwind address
;   100      0161  1         MIN_DIGITS: SIGNED                              ! Minimum number of digits
;   101      0162  1     ): NOVALUE =
;   102      0163  1
;   103      0164  1 !++
;   104      0165  1 ! FUNCTIONAL DESCRIPTION:
;   105      0166  1 !
;   106      0167  1 !       This procedure writes a base 2 representation of a value to the
;   107      0168  1 !       specified textfile.
;   108      0169  1 !
;   109      0170  1 ! CALLING SEQUENCE:
;   110      0171  1 !
;   111      0172  1 !       CALL PAS$WRITE_BIN (PFV.mr.r, NBITS.rl.v, VALUE.rz.r, TOTAL_WIDTH.rl.v
;   112      0173  1 !                       [, [ERROR.j.r] [, MIN_DIGITS.rl.v ]])
;   113      0174  1 !
;   114      0175  1 ! FORMAL PARAMETERS:
;   115      0176  1 !
;   116      0177  1 !       PFV             - The Pascal File Variable (PFV) passed by reference.
;   117      0178  1 !                         The structure of the PFV is defined in PASPFV.REQ.
;   118      0179  1 !
;   119      0180  1 !       NBITS           - The size of VALUE in bits.
;   120      0181  1 !
;   121      0182  1 !       VALUE           - The address of the value to write.
;   122      0183  1 !
;   123      0184  1 !       TOTAL_WIDTH     - Total field width.
;   124      0185  1 !
;   125      0186  1 !       ERROR           - Optional.  Address to unwind to if an error occurs.
;   126      0187  1 !
;   127      0188  1 !       MIN_DIGITS      - Optional.  The minimum number of digits to appear
;   128      0189  1 !                         in the result.  Defaults to the minimum number of
;   129      0190  1 !                         digits to represent every bit of the value.
;   130      0191  1 !
;   131      0192  1 ! IMPLICIT INPUTS:
;   132      0193  1 !
;   133      0194  1 !       NONE
;   134      0195  1 !
;   135      0196  1 ! IMPLICIT OUTPUTS:
;   136      0197  1 !
;   137      0198  1 !       NONE
;   138      0199  1 !
;   139      0200  1 ! ROUTINE VALUE:
;   140      0201  1 !
;   141      0202  1 !       NONE
;   142      0203  1 !
;   143      0204  1 ! SIDE EFFECTS:
;   144      0205  1 !
;   145      0206  1 !       If the file is the standard file INPUT or OUTPUT, it is implicitly opened.
;   146      0207  1 !
;   147      0208  1 ! SIGNALLED ERRORS:
;   148      0209  1 !
;   149      0210  1 !       NEGWIDDIG - negative field width or digits specification is not allowed
```

```
1 !        LINTOOLON - line too long
1 !
1 !--
1
2     BEGIN
2
2     LOCAL
2         FCB: REF $PAS$FCB_CONTROL_BLOCK,                    ! File control block
2         ACTUAL_DIGITS,                                      ! Number of digits actually used
2         ACTUAL_NBITS,                                       ! Value size actually used
2         DESCR: BLOCK [8, BYTE],                             ! String descriptor
2         PFV_ADDR: VOLATILE,                                 ! Enable argument
2         UNWIND_ACT: VOLATILE,                               ! Enable argument
2         ERROR_ADDR: VOLATILE;                               . Enable argument
2
2     LITERAL
2         M_SIZE_IN_BITS = %X'04';                            ! Flags argument for
2                                                             ! convert routine
2     BUILTIN
2         ACTUALCOUNT;
2
2     ENABLE
2         PAS$$IO_HANDLER (PFV_ADDR, UNWIND_ACT, ERROR_ADDR);     ! Enable error handler
2
2     !+
2     ! Get ERROR parameter, if present.
2     !-
2
2     IF ACTUALCOUNT () GEQU 5
2     THEN
2         ERROR_ADDR = .ERROR;            ! Set unwind address
2
2     PFV_ADDR = PFV [PFV$R_PFV];         ! Set PFV address
2
2     !+
2     ! Validate PFV and get PFV.
2     !-
2
2     PAS$$VALIDATE_PFV (PFV [PFV$R_PFV]; FCB);
2
2     !+
2     ! Set unwind action to unlock file.
2     !-
2
2     UNWIND_ACT = PAS$K_UNWIND_UNLOCK;
2
2     !+
2     ! Do common initialization.
2     !-
2
2     PAS$$INIT_WRITE (PFV [PFV$R_PFV], FCB [FCB$R_FCB]; FCB);
2
2     !+
2     ! Set initial values for conversion.
2     !-
2
2     ACTUAL_NBITS = .NBITS;
```

```
;  207        0268  2      ACTUAL_DIGITS = .ACTUAL_NBITS;
;  208        0269  2
;  209        0270  2      !+
;  210        0271  2      ! Create result string descriptor with actual width.
;  211        0272  2      !-
;  212        0273  2
;  213        0274  2      DESCR [DSC$B_CLASS] = DSC$K_CLASS_S;
;  214        0275  2      DESCR [DSC$B_DTYPE] = DSC$K_DTYPE_T;
;  215        0276  2      DESCR [DSC$A_POINTER] = .FCB [FCB$A_RECORD_CUR];
;  216        0277  2      IF ACTUALCOUNT () GEQU 6
;  217        0278  2      THEN
;  218        0279  3          BEGIN
;  219        0280  3          ACTUAL_DIGITS = .MIN_DIGITS;
;  220        0281  3          IF .ACTUAL_DIGITS LSS 0
;  221        0282  3          THEN
;  222        0283  3              $PAS$IO_ERROR (PAS$_NEGWIDDIG,0);
;  223        0284  2          END;
;  224        0285  2      IF .TOTAL_WIDTH LSS 0
;  225        0286  2      THEN
;  226        0287  2          $PAS$IO_ERROR (PAS$_NEGWIDDIG,0);
;  227        0288  2      DESCR [DSC$W_LENGTH] = .TOTAL_WIDTH;
;  228        0289  2
;  229        0290  2      !+
;  230        0291  2      ! Will TOTAL_WIDTH truncate the value?
;  231        0292  2      !+
;  232        0293  2      IF .TOTAL_WIDTH LSSU .ACTUAL_NBITS
;  233        0294  2      THEN
;  234        0295  3          BEGIN
;  235        0296  3          !+
;  236        0297  3          ! Change value size and bit offset to cause a truncated result.
;  237        0298  3          !-
;  238        0299  3          ACTUAL_NBITS = .TOTAL_WIDTH;
;  239        0300  2          END;
;  240        0301  2
;  241        0302  2      IF .ACTUAL_DIGITS GTR .TOTAL_WIDTH
;  242        0303  2      THEN
;  243        0304  2          ACTUAL_DIGITS = .TOTAL_WIDTH;
;  244        0305  2
;  245        0306  2      !+
;  246        0307  2      ! See if field will fit in record.
;  247        0308  2      !-
;  248        0309  2
;  249        0310  3      BEGIN
;  250        0311  3      LOCAL
;  251        0312  3          EXTRA;                      ! Extra characters past end of line
;  252        0313  3      EXTRA = (.FCB [FCB$A_RECORD_CUR] + .TOTAL_WIDTH) - .FCB [FCB$A_RECORD_END];
;  253        0314  3      IF .EXTRA GTR 0
;  254        0315  3      THEN
;  255        0316  3          $PAS$IO_ERROR (PAS$_LINTOOLON,1,.EXTRA);
;  256        0317  2      END;
;  257        0318  2
;  258        0319  2      !+
;  259        0320  2      ! Do the conversion.  It can't fail.
;  260        0321  2      !-
;  261        0322  2
;  262        0323  2      OTS$CVT_L_TB (.VALUE, DESCR, .ACTUAL_DIGITS, .ACTUAL_NBITS,
;  263        0324  2                    M_SIZE_IN_BITS);
;  264          0325  2
;  265          0326  2      FCB [FCB$A_RECORD_CUR] = .FCB [FCB$A_RECORD_CUR] + .DESCR [DSC$W_LENGTH];
;  266          0327  2
;  267          0328  2      !+
;  268          0329  2      ! Call WRITE epilogue routine to move the last character written to the
;  269          0330  2      ! user's buffer and to unlock the file variable.
;  270          0331  2      !-
;  271          0332  2
;  272          0333  2      PAS$$END_WRITE (PFV [PFV$R_PFV], FCB [FCB$R_FCB]);
;  273          0334  2
;  274          0335  2      RETURN;
;  275          0336  2
;  276          0337  1      END;                                          ! End of routine PAS$WRITE_BIN


                                                                  .TITLE  PAS$WRITE_BIN Write value in base 2
                                                                  .IDENT  \1-002\

                                                                  .EXTRN  PAS$WRITE_BIN, PAS$WRITEV_BIN
                                                                  .EXTRN  PAS$$IO_HANDLER
                                                                  .EXTRN  PAS$$VALIDATE_PFV
                                                                  .EXTRN  PAS$$INIT_WRITE
                                                                  .EXTRN  PAS$$SIGNAL, PAS$K_NEGWIDDIG
                                                                  .EXTRN  PAS$K_LINTOOLON
                                                                  .EXTRN  OTS$CVT_L_TB, PAS$$END_WRITE

                                                                  .PSECT  _PAS$CODE,NOWRT,  SHR,  PIC,2

                                     01FC 00000                   .ENTRY  PAS$WRITE_BIN, Save R2,R3,R4,R5,R6,R7,R8         ; 0155
                  58 00000000G   00   9E 00002                    MOVAB   PAS$$SIGNAL, R8                                  ;
                  5E             10   C2 00009                    SUBL2   #16, SP                                          ;
                                 7E   D4 0000C                    CLRL    ERROR_ADDR                                       ; 0215
                           04    AE   7C 0000E                    CLRQ    UNWIND_ACT                                       ;
                  6D       009A  CF   DE 00011                    MOVAL   8$, (FP)                                         ;
                  05             6C   91 00016                    CMPB    (AP), #5                                         ; 0239
                                 04   1F 00019                    BLSSU   1$                                               ;
                  6E         14  AC   D0 0001B                    MOVL    ERROR, ERROR_ADDR                                ; 0241
                  56         04  AC   D0 0001F  1$:               MOVL    PFV, R6                                          ; 0243
            08    AE             56   D0 00023                    MOVL    R6, PFV_ADDR                                     ;
                     00000000G   00   16 00027                    JSB     PAS$$VALIDATE_PFV                                ; 0249
            04    AE             01   D0 0002D                    MOVL    #1, UNWIND_ACT                                   ; 0255
                     00000000G   00   16 00031                    JSB     PAS$$INIT_WRITE                                  ; 0261
                  54         08  AC   D0 00037                    MOVL    NBITS, ACTUAL_NBITS                              ; 0267
                  53             54   D0 0003B                    MOVL    ACTUAL_NBITS, ACTUAL_DIGITS                      ; 0268
            0E    AE       010E  8F   B0 0003E                    MOVW    #270, DESCR+2                                    ; 0275
            10    AE         EC  A7   D0 00044                    MOVL    -20(FCB), DESCR+4                                ; 0276
                  06             6C   91 00049                    CMPB    (AP), #6                                         ; 0277
                                 06   1F 0004C                    BLSSU   2$                                               ;
                  53         18  AC   D0 0004E                    MOVL    MIN_DIGITS, ACTUAL_DIGITS                        ; 0280
                                 06   19 00052                    BLSS    3$                                               ; 0281
                  52         10  AC   D0 00054  2$:               MOVL    TOTAL_WIDTH, R2                                  ; 0285
                                 0A   18 00058                    BGEQ    4$                                               ;
                                 7E   D4 0005A  3$:               CLRL    -(SP)                                            ; 0287
                  7E        00G  8F   9A 0005C                    MOVZBL  #PAS$K_NEGWIDDIG, -(SP)                          ;
                  68             02   FB 00060                    CALLS   #2, PAS$$SIGNAL                                  ;
                                      04 00063                    RET                                                      ;
            0C    AE             52   B0 00064  4$:               MOVW    R2, DESCR                                        ; 0288
```

```
                         54           52 D1 00068         CMPL    R2, ACTUAL_NBITS                  ; 0293
                                      03 1E 0006B         BGEQU   5$                                ;
                         54           52 D0 0006D         MOVL    R2, ACTUAL_NBITS                  ; 0299
                         52           53 D1 00070 5$:     CMPL    ACTUAL_DIGITS, R2                 ; 0302
                                      03 15 00073         BLEQ    6$                                ;
                         53           52 D0 00075         MOVL    R2, ACTUAL_DIGITS                 ; 0304
          50             52       EC  A7 C1 00078 6$:     ADDL3   -20(FCB), R2, R0                  ; 0313
                         50       F0  A7 C2 0007D         SUBL2   -16(FCB), EXTRA                   ;
                                      0C 15 00081         BLEQ    7$                                ; 0314
                                      50 DD 00083         PUSHL   EXTRA                             ; 0316
                                      01 DD 00085         PUSHL   #1                                ;
                         7E       00G 8F 9A 00087         MOVZBL  #PAS$K_LINTOOLON, -(SP)           ;
                         68           03 FB 0008B         CALLS   #3, PAS$$SIGNAL                   ;
                                         04 0008E         RET                                       ;
                                      04 DD 0008F 7$:     PUSHL   #4                                ; 0323
                                      18 BB 00091         PUSHR   #^M<R3,R4>                        ;
                                  18  AE 9F 00093         PUSHAB  DESCR                             ;
                                  0C  AC DD 00096         PUSHL   VALUE                             ;
             00000000G   00           05 FB 00099         CALLS   #5, OTS$CVT_L_TB                  ;
                         50       0C  AE 3C 000A0         MOVZWL  DESCR, R0                         ; 0326
                   EC    A7           50 C0 000A4         ADDL2   R0, -20(FCB)                      ;
                            00000000G 00 16 000A8         JSB     PAS$$END_WRITE                    ; 0333
                                         04 000AE         RET                                       ; 0337
                                       0000 000AF 8$:     .WORD   Save nothing                      ; 0215
                         50       08  AC D0 000B1         MOVL    8(AP), R0                         ;
                         50       04  A0 D0 000B5         MOVL    4(R0), R0                         ;
                                  EC  A0 9F 000B9         PUSHAB  ERROR_ADDR                        ;
                                  F0  A0 9F 000BC         PUSHAB  UNWIND_ACT                        ;
                                  F4  A0 9F 000BF         PUSHAB  PFV_ADDR                          ;
                                      03 DD 000C2         PUSHL   #3                                ;
                                      5E DD 000C4         PUSHL   SP                                ;
                         7E       04  AC 7D 000C6         MOVQ    4(AP), -(SP)                      ;
             00000000G   00           03 FB 000CA         CALLS   #3, PAS$$IO_HANDLER               ;
                                         04 000D1         RET                                       ;

; Routine Size:  210 bytes,    Routine Base:  _PAS$CODE + 0000


;  277          0338  1
;  278          0339  1 !<BLF/PAGE>
```

```
1 %SBTTL 'PAS$WRITEV_BIN - Write value in base 2 to string'
1 GLOBAL ROUTINE PAS$WRITEV_BIN (
1         MAX_LENGTH: WORD,                               ! Maximum length of string
1         STRING_LINE: REF VECTOR [, WORD],               ! String to write to
1         NBITS,                                          ! Number of bits in VALUE
1         VALUE: REF VECTOR [, BYTE],                     ! Value to write
1         TOTAL_WIDTH: SIGNED,                            ! Total field width
1         ERROR,                                          ! Error unwind address
1         MIN_DIGITS: SIGNED                              ! Minimum number of digits
1     ) : NOVALUE =
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This procedure writes a value in base 2 to the specified string.
1 !
1 ! CALLING SEQUENCE:
1 !
1 !       CALL PAS$WRITEV_BIN (MAX_LENGTH.rw.v, STRING_LINE.wvt.r,
1 !          NBITS.rl.v, VALUE.rz.v, TOTAL_WIDTH.rl.v
1 !          [, [ERROR.j.r] [, MIN_DIGITS.rl.v]])
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !       MAX_LENGTH      - The maximum length of STRING_LINE.
1 !
1 !       STRING_LINE     - A varying string to which the output will be appended.
1 !
1 !       NBITS           - Size of VALUE in bits.
1 !
1 !       VALUE           - The value to write.
1 !
1 !       TOTAL_WIDTH     - The width of the field to write.
1 !
1 !       ERROR           - Optional.  If specified, the address to unwind to
1 !                         in case of an error.
1 !
1 !       MIN_DIGITS      - Minimum number of digits to write.  Defaults to
1 !                         the minimum necessary to represent every bit of
1 !                         the value.
1 !
1 ! IMPLICIT INPUTS:
1 !
1 !       NONE
1 !
1 ! IMPLICIT OUTPUTS:
1 !
1 !       NONE
1 !
1 ! ROUTINE VALUE:
1 !
1 !       NONE
1 !
1 ! SIDE EFFECTS:
1 !
1 !       NONE
1 !
```

```
1 ! SIGNALLED ERRORS:
1 !
1 !       See PAS$WRITE_BIN
1 !
1 !--
1
2    BEGIN
2
2    LOCAL
2        PFV: $PAS$PFV_FILE_VARIABLE,    ! Pascal File Variable
2        ARG_LIST: VECTOR [7, LONG],     ! Argument list
2        PFV_ADDR: VOLATILE,             ! Enable argument
2        UNWIND_ACT: VOLATILE,           ! Enable argument
2        ERROR_ADDR: VOLATILE;           ! Enable argument
2
2    BUILTIN
2        ACTUALCOUNT;                    ! Count of arguments
2
2    ENABLE
2        PAS$$IO_HANDLER (PFV_ADDR, UNWIND_ACT, ERROR_ADDR);     ! Enable error handler
2
2    !+
2    ! Get ERROR parameter, if present.
2    !-
2
2    IF ACTUALCOUNT () GEQU 6
2    THEN
2        ERROR_ADDR = .ERROR;            ! Set unwind address
2
2    PFV_ADDR = PFV [PFV$R_PFV];         ! Set PFV address
2
2    !+
2    ! Set up ARG_LIST.
2    !-
2
2    ARG_LIST [0] = 4;                   ! Four arguments
2    ARG_LIST [1] = PFV [PFV$R_PFV];     ! PFV address
2    ARG_LIST [2] = .NBITS;              ! Number of bits
2    ARG_LIST [3] = VALUE [0];           ! Value to write
2    ARG_LIST [4] = .TOTAL_WIDTH;        ! Field width
2    IF ACTUALCOUNT () GEQU 7
2    THEN
3        BEGIN
3        ARG_LIST [0] = 6;               ! Add two more arguments
3        ARG_LIST [5] = 0;               ! Error address
3        ARG_LIST [6] = .MIN_DIGITS;     ! Minimum digits
2        END;
2
2    !+
2    ! Call PAS$$DO_WRITEV to do the work, giving it the address of
2    ! PAS$WRITE_BIN to call.
2    !-
2
2    PAS$$DO_WRITEV (PFV [PFV$R_PFV], .MAX_LENGTH, STRING_LINE [0], ARG_LIST,
2        PAS$WRITE_BIN);
2
2    RETURN;
```

```
;  394          0454  2
;  395          0455  1     END;                                          ! End of routine PAS$WRITEV_BIN



                                                          .EXTRN   PAS$$DO_WRITEV

                                     007C 00000           .ENTRY   PAS$WRITEV_BIN, Save R2,R3,R4,R5,R6              ; 0341
                   5E              34  C2 00002           SUBL2    #52, SP                                          ;
                                   7E  D4 00005           CLRL     ERROR_ADDR                                       ; 0403
                           04      AE  7C 00007           CLRQ     UNWIND_ACT                                       ;
                   6D    004F      CF  DE 0000A           MOVAL    3$, (FP)                                         ;
                   06              6C  91 0000F           CMPB     (AP), #6                                         ; 0422
                                   04  1F 00012           BLSSU    1$                                               ;
                   6E      18      AC  D0 00014           MOVL     ERROR, ERROR_ADDR                                ; 0424
              08   AE      28      AE  9E 00018 1$:       MOVAB    PFV, PFV_ADDR                                    ; 0426
              0C   AE              04  D0 0001D           MOVL     #4, ARG_LIST                                     ; 0432
              10   AE      28      AE  9E 00021           MOVAB    PFV, ARG_LIST+4                                  ; 0433
              14   AE      0C      AC  7D 00026           MOVQ     NBITS, ARG_LIST+8                                ; 0434
              1C   AE      14      AC  D0 0002B           MOVL     TOTAL_WIDTH, ARG_LIST+16                         ; 0436
                   07              6C  91 00030           CMPB     (AP), #7                                         ; 0437
                                   0C  1F 00033           BLSSU    2$                                               ;
              0C   AE              06  D0 00035           MOVL     #6, ARG_LIST                                     ; 0440
                           20      AE  D4 00039           CLRL     ARG_LIST+20                                      ; 0441
              24   AE      1C      AC  D0 0003C           MOVL     MIN_DIGITS, ARG_LIST+24                          ; 0442
                   55    FEE9      CF  9E 00041 2$:       MOVAB    PAS$WRITE_BIN, R5                                ; 0450
                   54      0C      AE  9E 00046           MOVAB    ARG_LIST, R4                                     ;
                   56      28      AE  9E 0004A           MOVAB    PFV, R6                                          ;
                   53      08      AC  D0 0004E           MOVL     STRING_LINE, R3                                  ;
                   52      04      AC  3C 00052           MOVZWL   MAX_LENGTH, R2                                   ;
                      00000000G    00  16 00056           JSB      PAS$$DO_WRITEV                                   ;
                                       04 0005C           RET                                                       ; 0455
                                     0000 0005D 3$:       .WORD    Save nothing                                     ; 0403
                   50      08      AC  D0 0005F           MOVL     8(AP), R0                                        ;
                   50      04      A0  D0 00063           MOVL     4(R0), R0                                        ;
                           C8      A0  9F 00067           PUSHAB   ERROR_ADDR                                       ;
                           CC      A0  9F 0006A           PUSHAB   UNWIND_ACT                                       ;
                           D0      A0  9F 0006D           PUSHAB   PFV_ADDR                                         ;
                                   03  DD 00070           PUSHL    #3                                               ;
                                   5E  DD 00072           PUSHL    SP                                               ;
                   7E      04      AC  7D 00074           MOVQ     4(AP), -(SP)                                     ;
         00000000G 00              03  FB 00078           CALLS    #3, PAS$$IO_HANDLER                              ;
                                       04 0007F           RET                                                       ;

; Routine Size:  128 bytes,    Routine Base:  _PAS$CODE + 00D2


;  396          0456  1
;  397          0457  1 !<BLF/PAGE>
;  399          0458  1 END                                              . End of module PAS$WRITE_BIN
;  400          0459  1
;  401          0460  0 ELUDOM




;                                         PSECT SUMMARY

;       Name                         Bytes                           Attributes

;  _PAS$CODE                              338  NOVEC,NOWRT,  RD ,  EXE,  SHR,  LCL,  REL,  CON,  PIC,ALIGN(2)




;                                  Library Statistics

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

;  _$255$DUA28:[SYSLIB]STARLET.L32;1               9776          6         0        581           00:01.0
;  _$255$DUA28:[PASRTL.OBJ]PASLIB.L32;1             427         97        22         33           00:00.4




;                                          COMMAND QUALIFIERS

;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACE/LIS=LIS$:PASWRIBIN/OBJ=OBJ$:PASWRIBIN MSRC$:PASWRIBIN/UPDATE-(FNH$:PASWRIBIN
;       )

; Size:          338 code + 0 data bytes
; Run Time:         00:08.9
; Elapsed Time:     00:29.3
; Lines/CPU Min:      3094
; Lexemes/CPU-Min: 14825
; Memory Used:  101 pages
; Compilation Complete
```

0297 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY