```
LLL              IIIIIIIIII    BBBBBBBBBBBB    RRRRRRRRRRRR    TTTTTTTTTTTTTTTT   LLL
LLL              IIIIIIIIII    BBBBBBBBBBBB    RRRRRRRRRRRR    TTTTTTTTTTTTTTTT   LLL
LLL              IIIIIIIIII    BBBBBBBBBBBB    RRRRRRRRRRRR    TTTTTTTTTTTTTTTT   LLL
LLL                  III           BBB    BBB  RRR        RRR          TTT        LLL
LLL                  III           BBB    BBB  RRR        RRR          TTT        LLL
LLL                  III           BBB    BBB  RRR        RRR          TTT        LLL
LLL                  III           BBB    BBB  RRR        RRR          TTT        LLL
LLL                  III           BBB    BBB  RRR        RRR          TTT        LLL
LLL                  III       BBBBBBBBBBBB    RRRRRRRRRRRR          TTT        LLL
LLL                  III       BBBBBBBBBBBB    RRRRRRRRRRRR          TTT        LLL
LLL                  III           BBB    BBB  RRR    RRR             TTT        LLL
LLL                  III           BBB    BBB  RRR    RRR             TTT        LLL
LLL                  III           BBB    BBB  RRR    RRR             TTT        LLL
LLL                  III           BBB    BBB  RRR        RRR          TTT        LLL
LLL                  III           BBB    BBB  RRR        RRR          TTT        LLL
LLL                  III           BBB    BBB  RRR        RRR          TTT        LLL
LLLLLLLLLLLLLLL   IIIIIIIIII    BBBBBBBBBBBB    RRR        RRR          TTT        LLLLLLLLLLLLLLL
LLLLLLLLLLLLLLL   IIIIIIIIII    BBBBBBBBBBBB    RRR        RRR          TTT        LLLLLLLLLLLLLLL
LLLLLLLLLLLLLLL   IIIIIIIIII    BBBBBBBBBBBB    RRR        RRR          TTT        LLLLLLLLLLLLLLL
```

```
  000000   TTTTTTTTTT   SSSSSSSS    CCCCCCCC  VV        VV  TTTTTTTTTT  PPPPPPPP      GGGGGGGG
  000000   TTTTTTTTTT   SSSSSSSS    CCCCCCCC  VV        VV  TTTTTTTTTT  PPPPPPPP      GGGGGGGG
00      00     TT     SS          CC           VV      VV       TT      PP      PP  GG
00      00     TT     SS          CC           VV      VV       TT      PP      PP  GG
00      00     TT     SS          CC           VV      VV       TT      PP      PP  GG
00      00     TT     SS          CC           VV      VV       TT      PP      PP  GG
00      00     TT     SSSSSS      CC           VV      VV       TT      PPPPPPPP    GG
00      00     TT     SSSSSS      CC           VV      VV       TT      PPPPPPPP    GG
00      00     TT         SS      CC           VV      VV       TT      PP          GG  GGGGGG
00      00     TT         SS      CC           VV      VV       TT      PP          GG  GGGGGG
00      00     TT         SS      CC             VV  VV         TT      PP          GG      GG
00      00     TT         SS      CC             VV  VV         TT      PP          GG      GG
  000000       TT     SSSSSSSS    CCCCCCCC        VV            TT      PP            GGGGGG
  000000       TT     SSSSSSSS    CCCCCCCC        VV            TT      PP            GGGGGG

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
```

```
                    0000    1        .TITLE  OTS$CVTPG_R9    Convert Packed to G floating
                    0000    2        .IDENT  /1-001/         ; File: OTSCVTPG.MAR Edit: PLL1001
                    0000    3
                    0000    4
                    0000    5    ;**********************************************************************
                    0000    6    ;*                                                                    *
                    0000    7    ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                            *
                    0000    8    ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.             *
                    0000    9    ;*  ALL RIGHTS RESERVED.                                               *
                    0000   10    ;*                                                                    *
                    0000   11    ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
                    0000   12    ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE *
                    0000   13    ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER *
                    0000   14    ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
                    0000   15    ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY *
                    0000   16    ;*  TRANSFERRED.                                                       *
                    0000   17    ;*                                                                    *
                    0000   18    ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE *
                    0000   19    ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT *
                    0000   20    ;*  CORPORATION.                                                       *
                    0000   21    ;*                                                                    *
                    0000   22    ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS *
                    0000   23    ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.            *
                    0000   24    ;*                                                                    *
                    0000   25    ;*                                                                    *
                    0000   26    ;**********************************************************************
                    0000   27
                    0000   28    ; FACILITY: LANGUAGE INDEPENDENT SUPPORT
                    0000   29    ;++
                    0000   30    ; ABSTRACT:
                    0000   31    ;       This module contains the routine that converts packed numbers
                    0000   32    ;       to g floating.
                    0000   33    ;
                    0000   34    ;
                    0000   35    ;--
                    0000   36    ;
                    0000   37    ; VERSION: 1
                    0000   38    ;
                    0000   39    ; HISTORY:
                    0000   40    ;
                    0000   41    ; AUTHOR:
                    0000   42    ;       Pamela Levesque, 18-Jan-1982
                    0000   43    ;
                    0000   44    ; MODIFIED BY:
                    0000   45    ;
                    0000   46    ;
                    0000   47    ;
```

OTS$CVTPG_R9
1-001

I 2

Convert Packed to G floating          16-SEP-1984 00:26:00  VAX/VMS Macro V04-00          Page  2
HISTORY  ; Detailed Current Edit History  6-SEP-1984 11:13:20  [LIBRTL.SRC]OTSCVTPG.MAR;1       (2)

```
0000    49              .SBTTL  HISTORY          ; Detailed Current Edit History
0000    50
0000    51
0000    52 ; Edit History for Version 1 of OTSCVTPG
0000    53 ;
0000    54 ; 1-001 - Original.  PLL 18-Jan-1982
0000    55
```

OTS$CVTPG_R9
1-001

J 2

Convert Packed to G floating          16-SEP-1984 00:26:00   VAX/VMS Macro V04-00        Page  3
DECLARATIONS                          6-SEP-1984 11:13:20  [LIBRTL.SRC]OTSCVTPG.MAR;1         (3)

```
        0000        57              .SBTTL  DECLARATIONS
        0000        58
        0000        59      ;
        0000        60      ; INCLUDE FILES:
        0000        61      ;
        0000        62              $DSCDEF
        0000        63
        0000        64      ;
        0000        65      ; EXTERNAL SYMBOLS:
        0000        66              .DSABL  GBL             ; Prevent undeclared symbols from being
        0000        67                                      ; automatically global
        0000        68
        0000        69              .EXTRN  OTS$CVT_T_G     ; D, E, F, G Conversion routine
        0000        70      ;
        0000        71
        0000        72      ;
        0000        73      ; MACROS:
        0000        74      ;       NONE
        0000        75      ;
        0000        76
        0000        77      ;
        0000        78      ; PSECT DECLARATIONS:
    00000000        79              .PSECT  _OTS$CODE       PIC, SHR, LONG, EXE, NOWRT,-
        0000        80                                      USR, CON, REL, LCL, RD
        0000        81
        0000        82      ;
        0000        83      ; EQUATED SYMBOLS:
        0000        84      ;       NONE
        0000        85      ;
        0000        86
        0000        87      ;
        0000        88      ; OWN STORAGE:
        0000        89      ;       NONE
        0000        90      ;
```

OTS$CVTPG_R9
1-001

K 2

Convert Packed to G floating
OTS$CVTPG_R9

16-SEP-1984 00:26:00  VAX/VMS Macro V04-00      Page  4
6-SEP-1984 11:13:20  [LIBRTL.SRC]OTSCVTPG.MAR;1      (4)

```
                                     0000    92              .SBTTL  OTS$CVTPG_R9
                                     0000    93
                                     0000    94  ;++
                                     0000    95  ; FUNCTIONAL DESCRIPTION:
                                     0000    96  ;
                                     0000    97  ;       Converts packed numbers to g floating.
                                     0000    98  ;
                                     0000    99  ; CALLING SEQUENCE:
                                     0000   100  ;
                                     0000   101  ;       JSB OTS$CVTPG_R9 (scale.rl.v, srclen.rl.v, src.rp.r, dst.wg.r)
                                     0000   102  ;
                                     0000   103  ;       Arguments are passed in R6, R7, R8 and R9.
                                     0000   104  ;
                                     0000   105  ; INPUT PARAMETERS:
                                     0000   106  ;
                                     0000   107  ;       SCALE.rl.v                 The power of ten by which the internal
                                     0000   108  ;                                  representation of the source must be
                                     0000   109  ;                                  multiplied to scale the same as the
                                     0000   110  ;                                  internal representation of the dest.
                                     0000   111  ;       SRCLEN.rl.v                The number of digits in the source
                                     0000   112  ;       SRC.rp.r                   The number to be converted
                                     0000   113  ;
                                     0000   114  ; IMPLICIT INPUTS:
                                     0000   115  ;
                                     0000   116  ;       All of the trap bits in the PSL are assumed off.
                                     0000   117  ;
                                     0000   118  ; OUTPUT PARAMETERS:
                                     0000   119  ;
                                     0000   120  ;       DST.wg.r                   The place to store the converted number
                                     0000   121  ;
                                     0000   122  ; IMPLICIT OUTPUTS:
                                     0000   123  ;
                                     0000   124  ;       NONE
                                     0000   125  ;
                                     0000   126  ; FUNCTION VALUE:
                                     0000   127  ;
                                     0000   128  ;       1 = SUCCESS, 0 = FAILURE
                                     0000   129  ;
                                     0000   130  ; SIDE EFFECTS:
                                     0000   131  ;
                                     0000   132  ;       Destroys registers R0 through R9.
                                     0000   133  ;
                                     0000   134  ;--
                                     0000   135
                                     0000   136
                                     0000   137  OTS$CVTPG_R9::
                        5E  28  C2   0000   138          SUBL2   #40,SP                   ; Space for temp string and result
        08 AE  1F  68   57  08       0003   139          CVTPS   R7,(R8),#31,8(SP)        ; Make a separate sign string
                                     0009   140  ;
                                     0009   141  ; Make a descriptor for the leading separate string.
                                     0009   142  ;
                                53   DD       0009   143          PUSHL   R3                       ; Address = temp string
                        7E  01  90   000B   144          MOVB    #DSC$K_CLASS_S,-(SP)     ; Class = static
                        7E  0E  90   000E   145          MOVB    #DSC$K_DTYPE_T,-(SP)     ; Data type = ASCII text
                        7E  20  B0   0011   146          MOVW    #32,-(SP)                ; Length = 32 bytes
                                     0014   147  ;
                                     0014   148  ; Now call the conversion routine.
```

```
                                  0014      149 ;
           7E    56    CE  0014      150               MNEGL    R6,-(SP)              ; Scale factor
                 00    DD  0017      151               PUSHL    #0                    ; Digits in fraction
                 10 AE 9F  0019      152               PUSHAB   16(SP)                ; Address of result area
                 0C AE 9F  001C      153               PUSHAB   12(SP)                ; Address of descriptor
   00000000'GF   04    FB  001F      154               CALLS    #4,G^OTS$CVT_T_G      ; Call the routine
           0C 50      E9   0026      155               BLBC     R0,2$                 ; Failure, must be overflow
     69 08 AE 50FD        0029      156               MOVG     8(SP),(R9)            ; Store result
           50 01    D0    002E      157               MOVL     #1,R0                 ; Indicate success
           5E 30    C0    0031      158 1$:            ADDL2    #48,SP                ; Delete stack temps
                    05    0034      159               RSB                            ; Return
                          0035      160 ;
                          0035      161 ; Come here on overflow to store the reserved operand.
                          0035      162 ;
     69 01   0F 79        0035      163 2$:            ASHQ     #15,#1,(R9)           ; Store reserved operand
           50    D4       0039      164               CLRL     R0                    ; Indicate failure
           F4    11       003B      165               BRB      1$                    ; Delete stack temps and return
                          003D      166 ;
                          003D      167               .END
```

```
DSC$K_CLASS_S                    = 00000001
DSC$K_DTYPE_T                    = 0000000E
OTS$CVTPG_R9                       00000000 RG    02
OTS$CVT_T_G                        ******** X     00
```

```
                              +-----------------+
                              ! Psect synopsis  !
                              +-----------------+
```

| PSECT name | Allocation | | | PSECT No. | | Attributes | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| .  ABS  . | 00000000 | ( | 0.) | 00 ( | 0.) | NOPIC | USR | CON | ABS | LCL | NOSHR | NOEXE | NORD | NOWRT | NOVEC | BYTE |
| $ABS$ | 00000000 | ( | 0.) | 01 ( | 1.) | NOPIC | USR | CON | ABS | LCL | NOSHR | EXE | RD | WRT | NOVEC | BYTE |
| _OTS$CODE | 0000003D | ( | 61.) | 02 ( | 2.) | PIC | USR | CON | REL | LCL | SHR | EXE | RD | NOWRT | NOVEC | LONG |

```
                           +-----------------------+
                           ! Performance indicators !
                           +-----------------------+
```

| Phase | Page faults | CPU Time | Elapsed Time |
|---|---|---|---|
| Initialization | 30 | 00:00:00.05 | 00:00:00.96 |
| Command processing | 110 | 00:00:00.33 | 00:00:03.29 |
| Pass 1 | 132 | 00:00:01.14 | 00:00:05.50 |
| Symbol table sort | 0 | 00:00:00.09 | 00:00:00.10 |
| Pass 2 | 42 | 00:00:00.29 | 00:00:01.27 |
| Symbol table output | 2 | 00:00:00.01 | 00:00:00.01 |
| Psect synopsis output | 2 | 00:00:00.01 | 00:00:00.01 |
| Cross-reference output | 0 | 00:00:00.00 | 00:00:00.00 |
| Assembler run totals | 320 | 00:00:01.92 | 00:00:11.14 |

The working set limit was 1050 pages.
8057 bytes (16 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 133 non-local and 2 local symbols.
167 source lines were read in Pass 1, producing 10 object records in Pass 2.
8 pages of virtual memory were used to define 7 macros.

```
                        +---------------------------+
                        ! Macro library statistics  !
                        +---------------------------+
```

| Macro library name | Macros defined |
|---|---|
| _$255$DUA28:[SYSLIB]STARLET.MLB;2 | 4 |

190 GETS were required to define 4 macros.

There were no errors, warnings or information messages.

MACRO/ENABLE=SUPPRESSION/DISABLE=(GLOBAL,TRACEBACK)/LIS=LIS$:OTSCVTPG/OBJ=OBJ$:OTSCVTPG MSRC$:OTSCVTPG/UPDATE=(ENH$:OTSCVTPG)

OTSCVTPD
LIS

OTSCVTTF
LIS

OTSCVTRFP
LIS

OTSCVTRT
LIS

OTSCVTTOL
LIS

OTSCVTPG
LIS

OTSCVTTR
LIS

OTSCVTRHP
LIS

OTSLINKAG
LIS

OTSCVTRDP
LIS

OTSCVTTIL
LIS

OTSCVTTLL
LIS

OTSCVTPF
LIS

OTSCVTRGP
LIS

OTSMOVE
LIS

OTSMSG
LIS

OTSCVTPH
LIS

OTSLUN
LIS