```
LLL                 IIIIIIIII     BBBBBBBBBBBB     RRRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
LLL                 IIIIIIIII     BBBBBBBBBBBB     RRRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
LLL                 IIIIIIIII     BBBBBBBBBBBB     RRRRRRRRRRRR     TTTTTTTTTTTTTTT  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
LLL                    III        BBBBBBBBBBBB     RRRRRRRRRRRR           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     IIIIIIIII     BBBBBBBBBBBB     RRR         RRR        TTT        LLLLLLLLLLLLLLL
LLLLLLLLLLLLLLL     IIIIIIIII     BBBBBBBBBBBB     RRR         RRR        TTT        LLLLLLLLLLLLLLL
LLLLLLLLLLLLLLL     IIIIIIIII     BBBBBBBBBBBB     RRR         RRR        TTT        LLLLLLLLLLLLLLL
```

FILEID**LIBPOLYG   C 1

```
LL            IIIIII    BBBBBBBB    PPPPPPPP      OOOOOO    LL          YY      YY    GGGGGGGG
LL            IIIIII    BBBBBBBB    PPPPPPPP      OOOOOO    LL          YY      YY    GGGGGGGG
LL              II      BB      BB  PP      PP  OO      OO  LL          YY      YY  GG
LL              II      BB      BB  PP      PP  OO      OO  LL          YY      YY  GG
LL              II      BB      BB  PP      PP  OO      OO  LL            YY  YY    GG
LL              II      BB      BB  PP      PP  OO      OO  LL            YY  YY    GG
LL              II      BBBBBBBB    PPPPPPPP    OO      OO  LL              YY      GG
LL              II      BBBBBBBB    PPPPPPPP    OO      OO  LL              YY      GG
LL              II      BB      BB  PP          OO      OO  LL              YY      GG  GGGGGG
LL              II      BB      BB  PP          OO      OO  LL              YY      GG  GGGGGG
LL              II      BB      BB  PP          OO      OO  LL              YY      GG      GG
LL              II      BB      BB  PP          OO      OO  LL              YY      GG      GG    ....
LLLLLLLLLL    IIIIII    BBBBBBBB    PP            OOOOOO    LLLLLLLLLL      YY        GGGGGG      ....
LLLLLLLLLL    IIIIII    BBBBBBBB    PP            OOOOOO    LLLLLLLLLL      YY        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
```

Table of contents

(2)	42	Edit History
(3)	48	DECLARATIONS
(4)	86	LIB$POLYG - Perform floating polynomial

```
0000            .TITLE  LIB$POLYG - Perform G floating polynomial calculation
0000            .IDENT  /1-003/         ; File: LIBPOLYG.MAR  Edit: SBL1003
0000
0000    ;
0000    ;****************************************************************************
0000    ;*                                                                          *
0000    ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
0000    ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
0000    ;*  ALL RIGHTS RESERVED.                                                    *
0000    ;*                                                                          *
0000    ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
0000    ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
0000    ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
0000    ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
0000    ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
0000    ;*  TRANSFERRED.                                                            *
0000    ;*                                                                          *
0000    ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
0000    ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
0000    ;*  CORPORATION.                                                            *
0000    ;*                                                                          *
0000    ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
0000    ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
0000    ;*                                                                          *
0000    ;*                                                                          *
0000    ;****************************************************************************
0000    ;
0000
0000    ;++
0000    ; FACILITY: General Utility Library
0000    ;
0000    ; ABSTRACT:
0000    ;
0000    ;       Perform G floating point polynomial calculation.
0000    ;
0000    ; ENVIRONMENT: User Mode, AST Reentrant
0000    ;
0000    ;--
0000    ; AUTHOR: Steven B. Lionel, CREATION DATE: 05-Feb-79
0000    ;
```

```
0000          .SBTTL  Edit History
0000      ; 1-001 - Original.  SBL 5-Feb-1979
0000      ; 1-002 - Fix comments.  SBL 31-July-1979
0000      ; 1-003 - Use local handler to insure that exceptions other than those documented
0000      ;         as statuses are resignalled.  SBL 25-Sept-1980
```

```
                   0000          .SBTTL  DECLARATIONS
                   0000  ;
                   0000  ; INCLUDE FILES:
                   0000  ;
                   0000          $CHFDEF                          ; Condition handling symbols
                   0000          $SSDEF                           ; System symbols
                   0000  ;
                   0000  ; EXTERNAL DECLARATIONS:
                   0000  ;
                   0000          .EXTRN  LIB$SIG_TO_RET           ; Library routine to convert
                   0000                                           ; a signal to an error return
                   0000                                           ; to caller of LIB$POLYG.
                   0000                                           ; R0 = signalled condition
                   0000
                   0000  ;
                   0000  ; MACROS:
                   0000  ;
                   0000
                   0000  ;
                   0000  ; EQUATED SYMBOLS:
                   0000  ;
                   0000
         00000004  0000          arg = 4                          ; argument
         00000008  0000          degree = 8                       ; degree of polynomial
         0000000C  0000          coeff = 12                       ; address of coefficient
         00000010  0000          result = 16                      ; result of polynomial
                   0000                                           ; table
                   0000  ;
                   0000  ; OWN STORAGE:
                   0000  ;
                   0000
                   0000  ;
                   0000  ; PSECT DECLARATIONS:
                   0000  ;
               00000000          .PSECT _LIB$CODE PIC, USR, CON, REL, LCL, SHR, -
                   0000                          EXE, RD, NOWRT, LONG
                   0000
```

```
0000            .SBTTL  LIB$POLYG - Perform floating polynomial
0000    ;++
0000    ; FUNCTIONAL DESCRIPTION:
0000    ;
0000    ;       LIB$POLYG provides the functionality of the VAX hardware
0000    ;       instruction POLYG to high level language users.
0000    ;
0000    ;       The third operand points to a table (array) of G double
0000    ;       precision floating point coefficients.  The coefficient of
0000    ;       the highest order term of the polynomial is pointed to
0000    ;       by the table address operand, i.e. the first table element.
0000    ;       The table is specified with lower order coefficients stored
0000    ;       at increasing addresses.
0000    ;
0000    ;       The evaluation is carried out by Horner's method, and the
0000    ;       result is stored at the location pointed to by the fourth
0000    ;       operand.  The result computed is:
0000    ;
0000    ;               if d = degree
0000    ;               and x = arg
0000    ;               result = C[0]+x*(C[1]+x*(C[2]+ ... x*C[d]))
0000    ;
0000    ;       The unsigned word degree operand specifies the highest
0000    ;       numbered coefficient to participate in the evaluation.
0000    ;
0000    ;       For further detail, refer to the VAX-11 Architecture
0000    ;       Handbook for the description of POLYx.
0000    ;
0000    ; CALLING SEQUENCE:
0000    ;
0000    ;       status.wlc.v = LIB$POLYG (arg.rg.r, degree.rw.r, coeff.rg.ra,
0000    ;                                 result.wg.r)
0000    ;
0000    ; INPUT PARAMETERS:
0000    ;
0000    ;       arg.rg.r        - argument, "x" in polynomial
0000    ;       degree.rw.r     - degree of polynomial (GEQ 0)
0000    ;       coeff.rg.ra     - table of coefficients, G double floating
0000    ;
0000    ; IMPLICIT INPUTS:
0000    ;
0000    ;       NONE
0000    ;
0000    ; OUTPUT PARAMETERS:
0000    ;
0000    ;       result.wg.r     - result of calculation
0000    ;
0000    ; IMPLICIT OUTPUTS:
0000    ;
0000    ;       NONE
0000    ;
0000    ; FUNCTION VALUE:
0000    ;
0000    ;       SS$_NORMAL      - successful execution
0000    ;       SS$_FLTOVF      - floating overflow
0000    ;       SS$_FLTUND      - floating underflow
0000    ;       SS$_ROPRAND     - reserved operand, see VAX Architecture
```

```
                      0000  ;                                   manual for more details
                      0000  ;
                      0000  ; SIDE EFFECTS:
                      0000  ;
                      0000  ;           All other exceptions are signalled.
                      0000  ;
                      0000  ;--
                      0000
                403C  0000          .ENTRY  LIB$POLYG, ^M<IV,R2,R3,R4,R5>   ; Entry point, enable int. ovf.
                      0002                                                  ; and save R2, R3, R4, R5
                      0002
     6D   16'AF   9E  0002          MOVAB   B^HANDLER, (FP)                 ; Set up handler to process
                      0006                                                  ; exceptions
                      0006
0C BC   08 BC   04 BC 55FD  0006    POLYG   @arg(AP), -                     ; perform polynomial
                      000E                  @degree(AP), -                  ; trap on exception to
                      000E                  @coeff(AP)                      ; handler which will
                      000E                                                  ; unwind a return error
                      000E                                                  ; condition in R0 to
                      000E                                                  ; caller of LIB$POLYD.
                      000E
  10 BC      50   7D  000E          MOVQ    R0, @result(AP)                 ; return value
                      0012
     50      01   9A  0012          MOVZBL  #1, R0                          ; success status code
                      0015
                  04  0015          RET                                     ; return
                      0016
                      0016
                      0016  HANDLER:
                0000  0016          .WORD   0
                      0018
                      0018  ;+
                      0018  ;   If the exception is one of the documented exceptions for this routine,
                      0018  ;   call LIB$SIG_TO_RET to return it as a status.  Otherwise, resignal.
                      0018  ;   Also, resignal if the depth is not zero.
                      0018  ;-
                      0018
     50   08 AC   D0  0018              MOVL    CHF$L_MCHARGLST(AP), R0     ; Get mechanism vector address
          08 A0   D5  001C              TSTL    CHF$L_MCH_DEPTH(R0)         ; Is depth zero?
             41   12  001F              BNEQ    90$                         ; If not, resignal
     51   04 AC   D0  0021              MOVL    CHF$L_SIGARGLST(AP), R1     ; Get signal vector address
     50   04 A1   D0  0025              MOVL    CHF$L_SIG_NAME(R1), R0      ; Get signalled condition
048C 8F      50   B1  0029              CMPW    R0, #SS$_FLTOVF             ; Compare conditions
             2A   13  002E              BEQL    10$                         ; If it matches, don't resignal
049C 8F      50   B1  0030              CMPW    R0, #SS$_FLTUND
             23   13  0035              BEQL    10$
0454 8F      50   B1  0037              CMPW    R0, #SS$_ROPRAND
             1C   13  003C              BEQL    10$
04C4 8F      50   B1  003E              CMPW    R0, #SS$_FLTUND_F
             08   12  0043              BNEQ    5$
04 A1   049C 8F   3C  0045              MOVZWL  #SS$_FLTUND, CHF$L_SIG_NAME(R1) ; Change fault code to trap code
             0D   11  004B              BRB     10$
04B4 8F      50   B1  004D  5$:         CMPW    R0, #SS$_FLTOVF_F
             0E   12  0052              BNEQ    90$
04 A1   048C 8F   3C  0054              MOVZWL  #SS$_FLTOVF, CHF$L_SIG_NAME(R1)
00000000'GF  6C   FA  005A  10$:        CALLG   (AP), G^LIB$SIG_TO_RET      ; Return signal as a status
                  04  0061              RET
```

```
        50   0918 8F   3C  0062   200 90$:   MOVZWL  #SS$_RESIGNAL, R0                  ; Resignal condition
                       04  0067   201        RET
                           0068   202
                           0068   203        .END
```

**

```
ARG                             = 00000004
CHF$L_MCHARGLST                 = 00000008
CHF$L_MCH_DEPTH                 = 00000008
CHF$L_SIGARGLST                 = 00000004
CHF$L_SIG_NAME                  = 00000004
COEFF                           = 0000000C
DEGREE                          = 00000008
HANDLER                           00000016 R      02
LIB$POLYG                         00000000 RG     02
LIB$SIG_TO_RET                    ********    X   00
RESULT                          = 00000010
SS$_FLTOVF                      = 0000048C
SS$_FLTOVF_F                    = 000004B4
SS$_FLTUND                      = 0000049C
SS$_FLTUND_F                    = 000004C4
SS$_RESIGNAL                    = 00000918
SS$_ROPRAND                     = 00000454

                                        +----------------+
                                        ! 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
_LIB$CODE                       00000068  (  104.)  02 (  2.)    PIC   USR   CON   REL   LCL   SHR   EXE   RD   NOWRT NOVEC LONG

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

Phase                     Page faults    CPU Time       Elapsed Time
-----                     -----------    --------       ------------
Initialization                     29    00:00:00.05    00:00:00.94
Command processing                111    00:00:00.33    00:00:02.78
Pass 1                            190    00:00:02.72    00:00:12.36
Symbol table sort                   0    00:00:00.43    00:00:01.54
Pass 2                             52    00:00:00.56    00:00:03.09
Symbol table output                 4    00:00:00.02    00:00:00.02
Psect synopsis output               2    00:00:00.01    00:00:00.02
Cross-reference output              0    00:00:00.00    00:00:00.00
Assembler run totals              390    00:00:04.12    00:00:20.75

The working set limit was 1050 pages.
21671 bytes (43 pages) of virtual memory were used to buffer the intermediate code.
There were 30 pages of symbol table space allocated to hold 427 non-local and 3 local symbols.
203 source lines were read in Pass 1, producing 13 object records in Pass 2.
9 pages of virtual memory were used to define 8 macros.
```

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

Macro library name                  Macros defined
------------------                  --------------
_$255$DUA28:[SYSLIB]STARLET.MLB;2          5

486 GETS were required to define 5 macros.

There were no errors, warnings or information messages.

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

0209 AH-BT13A-SE DIGITAL EQUIPMENT CORPORATION
VAX/VMS V4.0 CONFIDENTIAL AND PROPRIETARY

LIBPOLYG
LIS

LIBREMQHI
LIS

LIBSIGSTO
LIS

LIBRENAME
LIS

LIBSCANC
LIS

LIBRDOBJ
LIS

LIBRUNPRO
LIS

LIBSIGNAL
LIS

LIBPUTOUT
LIS

LIBREMQTI
LIS

LIBSIGRET
LIS

LIBSIMTRA
LIS

LIBSCOPY
LIS

LIBPOLYH
LIS

LIBREVERT
LIS