


```

LL      IIIIII  BBBB8888  PPPPPPP  000000  LL      YY      YY  FFFFFFFFFF
LL      IIIIII  BBBB8888  PPPPPPP  000000  LL      YY      YY  FFFFFFFFFF
LL      II      BB      BB  PP      PP  00      00  LL      YY      YY  FF
LL      II      BB      BB  PP      PP  00      00  LL      YY      YY  FF
LL      II      BB      BB  PP      PP  00      00  LL      YY      YY  FF
LL      II      BB      BB  PP      PP  00      00  LL      YY      YY  FF
LL      II      BB      BB  PP      PP  00      00  LL      YY      YY  FF
LL      II      BB      BB  PP      PP  00      00  LL      YY      YY  FF
LL      II      BB      BB  PP      PP  00      00  LL      YY      YY  FF
LL      II      BB      BB  PP      PP  00      00  LL      YY      YY  FF
LL      II      BB      BB  PP      PP  00      00  LL      YY      YY  FF
LLLLLLLLLLLL IIIIII  BBBB8888  PPPPPPP  000000  LLLLLLLLLL YY      YY  FFFFFFFFFF
LLLLLLLLLLLL IIIIII  BBBB8888  PPPPPPP  000000  LLLLLLLLLL YY      YY  FFFFFFFFFF

```

```

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

```

LIBSPOLYF
Table of contents

G 16

- Perform floating polynomial calculatio 16-SEP-1984 00:15:56 VAX/VMS Macro V04-00

Page 0

(2) 52
(3) 90

DECLARATIONS
LIBSPOLYF - Perform floating polynomial

```
0000 1 .TITLE LIBSPOLYF - Perform floating polynomial calculation
0000 2 .IDENT /1-005/ ; File: LIBPOLYF.MAR Edit: SBL1005
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
0000 29 :++
0000 30 : FACILITY: General Utility Library
0000 31 :
0000 32 : ABSTRACT:
0000 33 :
0000 34 : Perform single precision floating point polynomial calculation.
0000 35 :
0000 36 : ENVIRONMENT: User Mode, AST Reentrant
0000 37 :
0000 38 :--
0000 39 : AUTHOR: Steven B. Lionel, CREATION DATE: 05-Oct-78
0000 40 :
0000 41 : MODIFIED BY:
0000 42 :
0000 43 : SBL, 05-Oct-78: VERSION 1
0000 44 : 1-001 - Original
0000 45 : 1-002 - Put version number in standard format, with three digits in
0000 46 : the edit number. JBS 16-NOV-78
0000 47 : 1-003 - Add "" to the PSECT directive. JBS 21-DEC-78
0000 48 : 1-004 - Minor code improvements. SBL 05-Feb-79
0000 49 : 1-005 - Use local handler to insure that all exceptions other than those
0000 50 : documented as being statuses are resigalled. SBL 25-Sept-1980
```

```
0000 52      .SBTTL  DECLARATIONS
0000 53      :
0000 54      : INCLUDE FILES:
0000 55      :
0000 56      $CHFDEF      ; Condition handling macros
0000 57      $$$DEF      ; System symbols
0000 58      :
0000 59      : EXTERNAL DECLARATIONS:
0000 60      :
0000 61      .EXTRN  LIB$$SIG_TO_RET      ; Library routine to convert
0000 62      :                          ; a signal to an error return
0000 63      :                          ; to caller of LIBSPOLYF.
0000 64      :                          ; R0 = signalled condition
0000 65      :
0000 66      : MACROS:
0000 67      :
0000 68      :
0000 69      :
0000 70      :
0000 71      : EQUATED SYMBOLS:
0000 72      :
0000 73      :
00000004 0000 74      arg = 4      ; argument
00000008 0000 75      degree = 8   ; degree of polynomial
0000000C 0000 76      coeff = 12  ; address of coefficient
00000010 0000 77      result = 16 ; result of polynomial
0000 78      :                          ; table
0000 79      :
0000 80      : OWN STORAGE:
0000 81      :
0000 82      :
0000 83      :
0000 84      : PSECT DECLARATIONS:
0000 85      :
00000000 86      .PSECT _LIB$CODE PIC, USR, CON, REL, LCL, SHR, -
0000 87      EXE, RD, NOWRT, LONG
0000 88
```

```
0000 90 .SBTTL LIB$POLYF - Perform floating polynomial
0000 91 :++
0000 92 : FUNCTIONAL DESCRIPTION:
0000 93 :
0000 94 : LIB$POLYF provides the functionality of the VAX hardware
0000 95 : instruction POLYF to high level language users.
0000 96 :
0000 97 : The third operand points to a table (array) of single
0000 98 : precision floating point coefficients. The coefficient of
0000 99 : the highest order term of the polynomial is pointed to
0000 100 : by the table address operand, i.e. the first table element.
0000 101 : The table is specified with lower order coefficients stored
0000 102 : at increasing addresses.
0000 103 :
0000 104 : The evaluation is carried out by Horner's method, and the
0000 105 : result is stored at the location pointed to by the fourth
0000 106 : operand. The result computed is:
0000 107 :
0000 108 :     if d = degree
0000 109 :     and x = arg
0000 110 :     result = C[0]+x*(C[1]+x*(C[2]+ ... x*(C[d]))
0000 111 :
0000 112 : The unsigned word degree operand specifies the highest
0000 113 : numbered coefficient to participate in the evaluation.
0000 114 :
0000 115 : For further detail, refer to the VAX-11/780 Architecture
0000 116 : Handbook for the description of POLYF.
0000 117 :
0000 118 : CALLING SEQUENCE:
0000 119 :
0000 120 :     status.wlc.v = LIB$POLYF (arg.rf.r, degree.rw.r, coeff.rf.ra,
0000 121 :                             result.wf.r)
0000 122 :
0000 123 : INPUT PARAMETERS:
0000 124 :
0000 125 :     arg.rf.r           - argument, 'x' in polynomial
0000 126 :     degree.rw.r        - degree of polynomial (GEQ 0)
0000 127 :     coeff.rf.ra        - table of coefficients, floating
0000 128 :
0000 129 : IMPLICIT INPUTS:
0000 130 :
0000 131 :     NONE
0000 132 :
0000 133 : OUTPUT PARAMETERS:
0000 134 :
0000 135 :     result.wf.r        - result of calculation
0000 136 :
0000 137 : IMPLICIT OUTPUTS:
0000 138 :
0000 139 :     NONE
0000 140 :
0000 141 : FUNCTION VALUE:
0000 142 :
0000 143 :     SSS_NORMAL         - successful execution
0000 144 :     SSS_FLTOVF         - floating overflow
0000 145 :     SSS_FLTUNC         - floating underflow
0000 146 :     SSS_ROPRAND        - reserved operand, see VAX Architecture
```

```

                                manual for more details
0000 147 :
0000 148 :
0000 149 : SIDE EFFECTS:
0000 150 :
0000 151 : All other exceptions are signalled
0000 152 :
0000 153 :--
0000 154 :
400C 0000 155 .ENTRY LIBSPOLYF, ^M<IV,R2,R3> ; Entry point, enable int. ovf.
0002 156 ; and save R2, R3
0002 157
6D 15'AF 9E 0002 158 MOVAB B^HANDLER, (FP) ; Enable local handler to
0006 159 ; exceptions
0006 160
OC BC 08 BC 04 BC 55 0006 161 POLYF @arg(AP), - ; perform polynomial
000D 162 @degree(AP), - ; trap on exception to
000D 163 @coeff(AP) ; handler which will
000D 164 ; unwind a return error
000D 165 ; condition in R0 to
000D 166 ; caller of LIBSPOLYF.
000D 167
10 BC 50 D0 000D 168 MOVL R0, @result(AP) ; return value
0011 169
50 01 9A 0011 170 MOVZBL #1, R0 ; success status code
0014 171
04 0014 172 RET ; return
0015 173
0015 174 HANDLER:
0000 0015 175 .WORD 0
0017 176
0017 177 :+
0017 178 : If the exception is one of the documented exceptions for this routine,
0017 179 : call LIB$SIG_TO_RET to return it as a status. Otherwise, resignal.
0017 180 : Also, resignal if the depth is not zero.
0017 181 :-
0017 182
50 08 AC D0 0017 183 MOVL CHF$M_MCHARGLST(AP), R0 ; Get mechanism vector address
08 A0 D5 001B 184 TSTL CHF$M_MCH_DEPTH(R0) ; Is depth zero?
41 12 001E 185 BNEQ 90$ ; If not, resignal
51 04 AC D0 0020 186 MOVL CHF$M_SIGARGLST(AP), R1 ; Get signal vector address
50 04 A1 D0 0024 187 MOVL CHF$M_SIG_NAME(R1), R0 ; Get signalled condition
048C 8F 50 B1 0028 188 CMPW R0, #SS$FLTUVF ; Compare conditions
2A 13 002D 189 BEQL 10$ ; If it matches, don't resignal
049C 8F 50 B1 002F 190 CMPW R0, #SS$FLTUND
23 13 0034 191 BEQL 10$
0454 8F 50 B1 0036 192 CMPW R0, #SS$ROPRAND
1C 13 003B 193 BEQL 10$
04C4 8F 50 B1 003D 194 CMPW R0, #SS$FLTUND_F
08 12 0042 195 BNEQ 5$
04 A1 049C 8F 3C 0044 196 MOVZWL #SS$FLTUND, CHF$M_SIG_NAME(R1) ; Change fault code to trap code
0D 11 004A 197 BRB 10$
04B4 8F 50 B1 004C 198 5$: CMPW R0, #SS$FLTUVF_F
0E 12 0051 199 BNEQ 90$
04 A1 048C 8F 3C 0053 200 MOVZWL #SS$FLTUVF, CHF$M_SIG_NAME(R1)
00000000'GF 6C FA 0059 201 10$: CALLG (AP), G^LIB$SIG_TO_RET ; Return signal as a status
04 0060 202 RET
50 0918 8F 3C 0061 203 90$: MOVZWL #SS$RESIGNAL, R0 ; Resignal condition

```

LIBSPOLYF
1-005

L 16

- Perform floating polynomial calculatio 16-SEP-1984 00:15:56 VAX/VMS Macro V04-00 Page 5
LIBSPOLYF - Perform floating polynomial 6-SEP-1984 11:09:46 [LIBRTL.SRC]LIBPOLYF.MAR;1 (3)

04	0066	204	RET
	0067	205	
	0067	206	
	0067	207	.END

LIBSPOLYF
Symbol table

M 16
- Perform floating polynomial calculation 16-SEP-1984 00:15:56 VAX/VMS Macro V04-00
6-SEP-1984 11:09:46 [LIBRTL.SRC]LIBPOLYF.MAR;1

Page 6
(3)

```

ARG = 00000004
CHFSL_MCHARGLST = 00000008
CHFSL_MCH_DEPTH = 00000008
CHFSL_SIGARGLST = 00000004
CHFSL_SIG_NAME = 00000004
COEFF = 0000000C
DEGREE = 00000008
HANDLER = 00000015 R 02
LIBSPOLYF = 00000000 RG 02
LIBSSIG_TO_RET = ***** X 00
RESULT = 00000010
SS$FLTUVF = 0000048C
SS$FLTUVF_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
\$ABSS	00000000 (0.)	01 (1.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
_LIB\$CODE	00000067 (103.)	02 (2.)	PIC USR CON REL LCL SHR EXE RD NOWRT NOVEC LONG

! Performance indicators !

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.04	00:00:01.36
Command processing	106	00:00:00.28	00:00:02.10
Pass 1	189	00:00:02.65	00:00:13.16
Symbol table sort	0	00:00:00.42	00:00:01.18
Pass 2	54	00:00:00.59	00:00:01.32
Symbol table output	4	00:00:00.02	00:00:00.02
Psect synopsis output	2	00:00:00.02	00:00:00.02
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	386	00:00:04.02	00:00:19.16

The working set limit was 1050 pages.
21690 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.
207 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

_S255SDUA28:[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\$:LIBPOLYF/OBJ=OBJ\$:LIBPOLYF MSRC\$:LIBPOLYF/UPDATE=(ENH\$:LIBPOLYF)

LIBPOLYG LIS

LIBREMCHI LIS

LIBSIGSTO LIS

LIBRENAME LIS

LIBSCANC LIS

LIBRDOBU LIS

LIBRINPRO LIS

LIBSIGNAL LIS

LIBPUTOUT LIS

LIBREMOTI LIS

LIBSIGRET LIS

LIBSIMTRA LIS

LIBPOLYH LIS

LIBREVERT LIS

LIBSCOPY LIS