

[illegible]

3

Sy

MT

MT
MT

MT

MT

MT
MTMT
MTMT
MTMT
MT

MT
MT

MT
MT

MT

MT

MT
MT

MI
MT

MT
MT

MT
MT

MT
MT

MT
MT

MT

111

227

MT

MT
MT

MT

MT

M1
M1MT
MT

M1

M1

10

1

10

100

10

1

[illegible]

The
623
The
479
1

(2) 54
(3) 77
(4) 124
(5) 187
(6) 279
(7) 342

HISTORY : Detailed Current Edit History
DECLARATIONS : Declarative Part of Module
MTHSDACOS - Standard Double Precision Floating DACOS
MTHSDACOS_R7 - Special DACOS routine
MTHSDACOSD - Standard Double Precision Floating DACOSD
MTHSDACOSD_R7 - Special DACOSD routine

```

0000 1      .TITLE  MTHSDACOS      ; Double Precision Floating Point Arc-cosine routine
0000 2      ; (DACOS,DACOSD)
0000 3      .IDENT  /1-008/        ; File: MTHDACOS.MAR  Edit: JCW1008
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     ; FACILITY: MATH LIBRARY
0000 30     ;++
0000 31     ; ABSTRACT:
0000 32     ;
0000 33     ; MTHSDACOS is a function which returns the double precision floating point
0000 34     ; arc-cosine in radians of its double precision floating point argument. The
0000 35     ; call is standard call-by-reference.
0000 36     ;
0000 37     ; MTHSDACOSD is a function which returns the double precision floating point
0000 38     ; arc-cosine in degrees of its double precision floating point argument. The
0000 39     ; call is standard call-by-reference.
0000 40     ;
0000 41     ;--
0000 42     ;
0000 43     ; VERSION: 01
0000 44     ;
0000 45     ; HISTORY:
0000 46     ; AUTHOR:
0000 47     ; Peter Yuo, 29-Jun-77: Version 01
0000 48     ;
0000 49     ; MODIFIED BY:
0000 50     ;
0000 51     ;
0000 52     ;

```



```
0000 54      .SBTTL HISTORY ; Detailed Current Edit History
0000 55
0000 56
0000 57 : ALGORITHMIC DIFFERENCES FROM FP-11/C ROUTINE: none
0000 58 :
0000 59 : Edit History for Version 01 of MTHSDACOS
0000 60 :
0000 61 : 0-2 MTH$ERROR changed to MTH$$SIGNAL.
0000 62 : MTH$... changed to MTH...
0000 63 : Changed error handling mechanism. Put error result in R0:R1 before
0000 64 : calling MTH$$SIGNAL in order to allow user modify error result.
0000 65 : 0-3 - Changed MOVD #0, to CLRD. TNH 10-Nov-77
0000 66 : 1-001 - Updated version number and copyright notice. JBS 16-NOV-78
0000 67 : 1-002 - Change MTH_INVARG to MTH$K_INVARGMAT. JBS 07-DEC-78
0000 68 : 1-003 - Add " to PSECT directive. JBS 22-DEC-78
0000 69 : 1-004 - Declare externals. SBL 17-May-1979
0000 70 : 1-005 - Change JSB entry to MTHSDACOS R7. RBG 28-Sept-1979
0000 71 : 1-006 - Added degree entry points. RNH 22-MAR-1981
0000 72 : 1-007 - Modified computation of 1 - X^2 to avoid loss of significance for
0000 73 : |X|>=1/2 RNH 02-Sept-1981
0000 74 : 1-008 - Corrected the floating representation of .5 from #4000 to #^X4000
0000 75 : JCW 14-Jul-1982
```

```
0000 77      .SBTTL  DECLARATIONS      ; Declarative Part of Module
0000 78
0000 79 :
0000 80 : INCLUDE FILES:
0000 81 :
0000 82 :
0000 83 :
0000 84 : EXTERNAL SYMBOLS:
0000 85 :
0000 86      .DSABL  GBL
0000 87      .EXTRN  MTH$DSQRT_R5
0000 88      .EXTRN  MTH$DATAN_R7
0000 89      .EXTRN  MTH$DATAND_R7
0000 90      .EXTRN  MTH$K_INVARGMAT
0000 91      .EXTRN  MTH$$SIGNAL
0000 92
0000 93 :
0000 94 : EQUATED SYMBOLS:
0000 95
00004080 0000 96      SD 1.0 = ^F1.0      ; 1.0
00000004 0000 97      value = 4      ; value.rd.r
0000 98
0000 99 :
0000 100 : MACROS:      none
0000 101 :
0000 102 : PSECT DECLARATIONS:
0000 103
00000000 0000 104      .PSECT  _MTH$CODE      PIC,SHR,LONG,EXE,NOWRT
0000 105      ; program section for math routines
0000 106 :
0000 107 : OWN STORAGE:  none
0000 108 :
0000 109 :
0000 110 : CONSTANTS:
0000 111 :
0000 112 :
0000 113 D_PI_OVER_2:
68C2 A221 0FDA 40C9 0000 114      .WORD  ^0040311, ^0007732, ^0121041, ^0064302
0008 115      ; PI/2
0008 116 D_PI:
68C2 A221 0FDA 4149 0008 117      .WORD  ^0040511, ^0007732, ^0121041, ^0064302
0010 118      ; PI
0010 119 D_90:
00000000 000043B4 0010 120      .LONG  ^X000043B4, ^X0      ; 90
0018 121 D_180:
00000000 00004434 0018 122      .LONG  ^X00004434, ^X0      ; 180
```



```
0020 124 .SBTTL MTH$DACOS - Standard Double Precision Floating DACOS
0020 125
0020 126
0020 127 :++
0020 128 : FUNCTIONAL DESCRIPTION:
0020 129 :
0020 130 : DACOS - double precision floating point function
0020 131 :
0020 132 : DACOS(X) is computed as:
0020 133 :
0020 134 : If X = 0, then DACOS(X) = PI/2.
0020 135 : If X = 1, then DACOS(X) = 0.
0020 136 : If X = -1, then DACOS(X) = PI.
0020 137 : If 0 < X < 1/2, then DACOS(X) = ATAN(SQRT(1-X**2)/X).
0020 138 : If 1/2 < X < 1, then DACOS(X) = ATAN(SQRT((1-X)*(1+X))/X).
0020 139 : If -1/2 < X < 0, then DACOS(X) = ATAN(SQRT(1-X**2)/X) + PI.
0020 140 : If -1 < X <= -1/2, then DACOS(X) = ATAN(SQRT((1-X)*(1+X))/X) + PI.
0020 141 : If 1 < |X|, error.
0020 142 :
0020 143 : CALLING SEQUENCE:
0020 144 :
0020 145 : DACOS.wd.v = MTH$DACOS(x.rd.r)
0020 146 :
0020 147 : INPUT PARAMETERS:
0020 148 :
00000004 0020 149 LONG = 4 ; define longword multiplier
00000004 0020 150 x = 1 * LONG ; Contents of x is the argument
0020 151 :
0020 152 : IMPLICIT INPUTS: none
0020 153 :
0020 154 : OUTPUT PARAMETERS:
0020 155 :
0020 156 : VALUE: double precision floating arc-cosine of the argument
0020 157 :
0020 158 : IMPLICIT OUTPUTS: none
0020 159 :
0020 160 : COMPLETION CODES: none
0020 161 :
0020 162 : SIDE EFFECTS:
0020 163 :
0020 164 : Signals: MTH$_INVARG if |X| > 1 with reserved operand in R0/R1 (copied
0020 165 : to the signal mechanism vector CHF$MCH_R0/R1 by LIB$SIGNAL).
0020 166 : Associated message is: "INVALID ARGUMENT". Result is reserved operand -0.0
0020 167 : unless a user supplied (or any) error handler changes CHF$MCH_R0/R1.
0020 168 :
0020 169 : NOTE: This procedure disables floating point underflow, enables integer
0020 170 : overflow.
0020 171 :
0020 172 : ---
0020 173 :
0020 174 :
0020 175 .ENTRY MTH$DACOS, ^M<IV, R2, R3, R4, R5, R6, R7>
0022 176 : standard call-by-reference entry
0022 177 : disable DV (and FU), enable IV
0022 178 MTH$FLAG_JACKET ; flag that this is a jacket procedure in
0022
6D 00000000'GF 9E 0022 MOVAB G^MTH$$JACKET_HND, (FP)
```

```

; Double Precision Floating Point Arc-co 16-SEP-1984 01:13:42 VAX/VMS Macro V04-00 Page 5
MTHSDACOS - Standard Double Precision F 6-SEP-1984 11:21:34 [MTHRTL.SRC]MTHDACOS.MAR;1 (4)

```

				0029			; set handler address to jacket
				0029			; handler
				0029			
				0029	179		; case of an error in routine
				0029	180		; If an error, convert signal to user PC
				0029	181		; and resignal
50	04	BC	70	0029	182	MOVD	@value(AP), R0
		01	10	002D	183	BSBB	MTH\$DACOS_R7
			04	002F	184	RET	; call special DACOS routine
				0030	185		; return with result in R0/R1


```
0030 187 .SBTTL MTHSDACOS_R7 - Special DACOS routine
0030 188
0030 189 ; Special DACOS - used by the standard routine and direct JSB call.
0030 190
0030 191 ; CALLING SEQUENCE:
0030 192 ; save anything needed in R0:R7
0030 193 ; MOV R0 ; input in R0/R1
0030 194 ; JSB MTHSDACOS_R7
0030 195 ; RSB ; return with result in R0/R1
0030 196
0030 197
0030 198 MTHSDACOS_R7:: ; special DACOS routine
0030 199 MTHSDACOS_R9:: ; Release 1 name
56 50 70 0030 200 MOV R0, R6 ; save X in R6/R7
05 12 0033 201 BNEQ TEST_FOR_1.0 ; branch if !X! > 0
0035 202
0035 203 ;
0035 204 ; X = 0
0035 205 ;
0035 206
50 C8 AF 70 0035 207 MOV D_PI_OVER_2, R0 ; R0/R1 = PI/2
05 05 0039 208 RSB ; return PI/2 if !X! = 0
003A 209
003A 210 ;
003A 211 ; 0 < !X!
003A 212 ;
003A 213
003A 214 TEST_FOR_1.0:
50 8000 8F AA 003A 215 BICW #X8000, R0 ; R0/R1 = !X!
08 50 71 003F 216 CMPD R0, S^#SD_1.0 ; compare !X! with 1.0
30 18 0042 217 BGEQ GEQ_TO_1.0 ; branch if !X! >= 1.0
0044 218
0044 219 ;
0044 220 ; 0 < !X! < 1.0
0044 221 ;
0044 222
50 4000 8F B1 0044 223 CMPW #X4000, R0 ; Check for possible loss of
0C 14 0049 224 BGTR 1$ ; significance
52 08 50 63 004B 225 SUBD3 R0, #1, R2 ; R2/R3 = 1 - X
50 08 60 004F 226 ADDD #1, R0 ; R0/R1 = 1 + X
50 52 64 0052 227 MULD R2, R0 ; R0/R1 = 1 - X^2
07 11 0055 228 BRB 2$ ; Join main flow
50 50 50 64 0057 229 1$: MULD2 R0, R0 ; R0/R1 = X**2
08 50 63 005A 230 SUBD3 R0, S^#SD_1.0, R0 ; R0/R1 = 1.0 - X**2
00000000 EF 16 005E 231 2$: JSB MTHSDSQRT_R5 ; R0/R1 = DSQRT(1-X**2)
50 56 66 0064 232 DIVD R6, R0 ; R0/R1 = DSQRT(1-X**2)/X
56 DD 0067 233 PUSHL R6 ; save sign of X for sign test
00000000 EF 16 0069 234 JSB MTHSDATAN_R7 ; R0/R1 = DATAN(DSQRT(1-X**2)/X)
56 8E D0 006F 235 MOVL (SP)+, R6 ; restore sign of X
04 11 0072 236 BRB TEST_SIGN ; branch to TEST_SIGN
0074 237
0074 238 ;
0074 239 ; 1 <= !X!
0074 240 ;
0074 241
0074 242 GEQ_TO_1.0:
0B 14 0074 243 BGTR ERROR ; branch to ERROR if !X! > 1.0
```

```
0076 244
0076 245
0076 246 ;
0076 247 ; !X! = 1.0
0076 248 ;
0076 249 ;
50 7C 0076 250 CLRD R0 ; R0/R1 = 0
0078 251
0078 252
0078 253 ;
0078 254 ; Test the sign of X in order to decide if add PI to the result
0078 255 ;
0078 256 ;
0078 257 TEST_SIGN:
56 73 0078 258 TSTD R6 ; test the sign of X
04 18 007A 259 BGEQ 10$ ; branch if X > 0
50 89 AF 60 007C 260 ADDD D_F1, R0 ; add PI to R0/R1 if X < 0
05 0080 261 10$: RSB ; return with result in R0/R1
0081 262
0081 263 ;
0081 264 ; 1 < !X!, error
0081 265 ;
0081 266 ;
7E 00 6E DD 0081 267 ERROR: PUSHL (SP) ; return PC from JSB routine
50 01 0F 9A 0083 268 MOVZBL #MTH$K_INVARGMAT, -(SP) ; condition value
0087 269 ASHQ #15, #T, R0 ; R0 = result = reserved operand -0.0
008B 270 ; goes to signal mechanism vector
008B 271 ; (CHF$MCH_R0/R1) so error handler
008B 272 ; can modify the result.
00000000'GF 02 FB 008B 273 CALLS #2, G^MTH$$SIGNAL ; signal error and use real user's PC
0092 274 ; independent of CALL vs JSB
05 0092 275 RSB ; return - R0 restored from CHF$MCH_R0/R1
0093 276
0093 277
```



```
0093 279 .SBTTL MTH$DACOSD - Standard Double Precision Floating DACOSD
0093 280
0093 281
0093 282 :++
0093 283 : FUNCTIONAL DESCRIPTION:
0093 284 :
0093 285 : DACOSD - double precision floating point function
0093 286 :
0093 287 : DACOSD(X) is computed as:
0093 288 :
0093 289 :     If X = 0, then DACOSD(X) = 90.
0093 290 :     If X = 1, then DACOSD(X) = 0.
0093 291 :     If X = -1, then DACOSD(X) = 180.
0093 292 :     If 0 < X < 1, then DACOSD(X) = ATAN(SQRT(1-X**2)/X).
0093 293 :     If 1/2 < X < 1, then DACOSD(X) = ATAN(SQRT((1-X)*(1+X))/X).
0093 294 :     If -1/2 < X < 0, then DACOSD(X) = ATAN(SQRT(1-X**2)/X) + 180.
0093 295 :     If -1 < X < -1/2, then DACOSD(X) = ATAN(SQRT((1-X)*(1+X))/X) + 180.
0093 296 :     If 1 < |X|, error.
0093 297 :
0093 298 : CALLING SEQUENCE:
0093 299 :
0093 300 :     DACOSD.wd.v = MTH$DACOSD(x.rd.r)
0093 301 :
0093 302 : INPUT PARAMETERS:
0093 303 :
0093 304 :     LONG = 4 ; define longword multiplier
0093 305 :     x = 1 * LONG ; Contents of x is the argument
0093 306 :
0093 307 : IMPLICIT INPUTS: none
0093 308 :
0093 309 : OUTPUT PARAMETERS:
0093 310 :
0093 311 :     VALUE: double precision floating arc-cosine of the argument
0093 312 :
0093 313 : IMPLICIT OUTPUTS: none
0093 314 :
0093 315 : COMPLETION CODES: none
0093 316 :
0093 317 : SIDE EFFECTS:
0093 318 :
0093 319 : Signals: MTH$_INVARG if |X| > 1 with reserved operand in R0/R1 (co180ed
0093 320 : to the signal mechanism vector CHF$MCH_R0/R1 by LIB$SIGNAL).
0093 321 : Associated message is: "INVALID ARGUMENT". Result is reserved operand -0.0
0093 322 : unless a user supplied (or any) error handler changes CHF$MCH_R0/R1.
0093 323 :
0093 324 : NOTE: This procedure disables floating point underflow, enables integer
0093 325 : overflow.
0093 326 :
0093 327 : ---
0093 328 :
0093 329 :
0093 330 .ENTRY MTH$DACOSD, ^M<IV, R2, R3, R4, R5, R6, R7>
0095 331 ; standard call-by-reference entry
0095 332 ; disable DV (and FU), enable IV
0095 333 MTH$FLAG_JACKET ; flag that this is a jacket procedure in
0095 334
0095 335 MOVAB G^MTH$$JACKET_HND, (FP)
```

00000004
00000004

40FC

6D 00000000'GF 9E

```

; Double Precision Floating Point Arc-co 16-SEP-1984 01:13:42 VAX/VMS Macro V04-00 Page 9
MTH$DACOSD - Standard Double Precision 6-SEP-1984 11:21:34 [MTHRTL.SRC]MTHDACOS.MAR;1 (6)

```

			009C			; set handler address to jacket
			009C			; handler
			009C			
			009C	334		; case of an error in routine
			009C	335		; If an error, convert signal to user PC
			009C	336		; and resignal
50	04	BC	70	009C	337	MOVD @value(AP), R0
		01	10	00A0	338	BSBB MTH\$DACOSD_R7
			04	00A2	339	RET
				00A3	340	

MT
1-

PC	Op	Op2	Op3	Op4	Op5	Op6	Op7	Op8	Op9	Op10	Op11	Op12	Op13	Op14	Op15	Op16	Op17	Op18	Op19	Op20	Op21	Op22	Op23	Op24	Op25	Op26	Op27	Op28	Op29	Op30	Op31	Op32	Op33	Op34	Op35	Op36	Op37	Op38	Op39	Op40	Op41	Op42	Op43	Op44	Op45	Op46	Op47	Op48	Op49	Op50	Op51	Op52	Op53	Op54	Op55	Op56	Op57	Op58	Op59	Op60	Op61	Op62	Op63	Op64	Op65	Op66	Op67	Op68	Op69	Op70	Op71	Op72	Op73	Op74	Op75	Op76	Op77	Op78	Op79	Op80	Op81	Op82	Op83	Op84	Op85	Op86	Op87	Op88	Op89	Op90	Op91	Op92	Op93	Op94	Op95	Op96	Op97	Op98	Op99	Op100	Op101	Op102	Op103	Op104	Op105	Op106	Op107	Op108	Op109	Op110	Op111	Op112	Op113	Op114	Op115	Op116	Op117	Op118	Op119	Op120	Op121	Op122	Op123	Op124	Op125	Op126	Op127	Op128	Op129	Op130	Op131	Op132	Op133	Op134	Op135	Op136	Op137	Op138	Op139	Op140	Op141	Op142	Op143	Op144	Op145	Op146	Op147	Op148	Op149	Op150	Op151	Op152	Op153	Op154	Op155	Op156	Op157	Op158	Op159	Op160	Op161	Op162	Op163	Op164	Op165	Op166	Op167	Op168	Op169	Op170	Op171	Op172	Op173	Op174	Op175	Op176	Op177	Op178	Op179	Op180	Op181	Op182	Op183	Op184	Op185	Op186	Op187	Op188	Op189	Op190	Op191	Op192	Op193	Op194	Op195	Op196	Op197	Op198	Op199	Op200	Op201	Op202	Op203	Op204	Op205	Op206	Op207	Op208	Op209	Op210	Op211	Op212	Op213	Op214	Op215	Op216	Op217	Op218	Op219	Op220	Op221	Op222	Op223	Op224	Op225	Op226	Op227	Op228	Op229	Op230	Op231	Op232	Op233	Op234	Op235	Op236	Op237	Op238	Op239	Op240	Op241	Op242	Op243	Op244	Op245	Op246	Op247	Op248	Op249	Op250	Op251	Op252	Op253	Op254	Op255	Op256	Op257	Op258	Op259	Op260	Op261	Op262	Op263	Op264	Op265	Op266	Op267	Op268	Op269	Op270	Op271	Op272	Op273	Op274	Op275	Op276	Op277	Op278	Op279	Op280	Op281	Op282	Op283	Op284	Op285	Op286	Op287	Op288	Op289	Op290	Op291	Op292	Op293	Op294	Op295	Op296	Op297	Op298	Op299	Op300	Op301	Op302	Op303	Op304	Op305	Op306	Op307	Op308	Op309	Op310	Op311	Op312	Op313	Op314	Op315	Op316	Op317	Op318	Op319	Op320	Op321	Op322	Op323	Op324	Op325	Op326	Op327	Op328	Op329	Op330	Op331	Op332	Op333	Op334	Op335	Op336	Op337	Op338	Op339	Op340	Op341	Op342	Op343	Op344	Op345	Op346	Op347	Op348	Op349	Op350	Op351	Op352	Op353	Op354	Op355	Op356	Op357	Op358	Op359	Op360	Op361	Op362	Op363	Op364	Op365	Op366	Op367	Op368	Op369	Op370	Op371	Op372	Op373	Op374	Op375	Op376	Op377	Op378	Op379	Op380	Op381	Op382	Op383	Op384	Op385	Op386	Op387	Op388	Op389	Op390	Op391	Op392	Op393	Op394	Op395	Op396	Op397	Op398	Op399	Op400	Op401	Op402	Op403	Op404	Op405	Op406	Op407	Op408	Op409	Op410	Op411	Op412	Op413	Op414	Op415	Op416	Op417	Op418	Op419
----	----	-----	-----	-----	-----	-----	-----	-----	-----	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------

```

00ED 399
00ED 400
00ED 401 ;
00ED 402 ; |X| = 1.0
00ED 403 ;
50 7C 00ED 404
00ED 405 10$: CLRD R0 ; R0/R1 = 0
00EF 406
00EF 407
00EF 408 ;
00EF 409 ; Test the sign of X in order to decide if add 180 to the result
00EF 410 ;
00EF 411 ;
00EF 412 D_TEST_SIGN:
56 73 00EF 413 TSTD R6 ; test the sign of X
05 18 00F1 414 BGEQ 10$ ; branch if X > 0
50 FF21 CF 60 00F3 415 ADDD D_180, R0 ; add 180 to R0/R1 if X < 0
05 00F8 416 10$: RSB ; return with result in R0/R1
00F9 417
00F9 418
00F9 419 .END

```


MTH\$DACOS
Symbol table

C 14
; Double Precision Floating Point Arc-co 16-SEP-1984 01:13:42 VAX/VMS Macro V04-00
6-SEP-1984 11:21:34 [MTHRTL.SRC]MTHDACOS.MAR;1

Page 12
(7)

```
D_180      00000018 R    01
D_90       00000010 R    01
D_GEQ_TO_1.0 000000E8 R    01
D_PI       00000008 R    01
D_PI_OVER 2 00000000 R    01
D_TEST_FOR 1.0 000000AE R    01
D_TEST_SIGN 000000EF R    01
ERROR      00000081 R    01
GEQ_TO_1.0 00000074 R    01
LONG       = 00000004
MTH$$JACKET_HND ***** X    01
MTH$$SIGNAL ***** X    00
MTH$DACOS  00000020 RG    01
MTH$DACOSD 00000093 RG    01
MTH$DACOSD R7 000000A3 RG    01
MTH$DACOS R7 00000030 RG    01
MTH$DACOS R9 00000030 RG    01
MTH$DATAND R7 ***** X    00
MTH$DATAN R7 ***** X    00
MTH$DSQRT R5 ***** X    00
MTH$K_INVARGMAT ***** X    00
SD_1.0     = 00004080
TEST_FOR 1.0 0000003A R    01
TEST_SIGN 00000078 R    01
VALUE      = 00000004
```

! Psect synopsis !

PSECT name	Allocation	PSECT No.	Attributes															
. ABS	00000000 (0.)	00 (0.)	NOPIC	USR	CON	ABS	LCL	NOSHR	NOEXE	NORD	NOWRT	NOVEC	BYTE					
_MTH\$CODE	000000F9 (249.)	01 (1.)	PIC	USR	CON	REL	LCL	SHR	EXE	RD	NOWRT	NOVEC	LONG					

! Performance indicators !

Phase	Page faults	CPU Time	Elapsed Time
Initialization	31	00:00:00.11	00:00:00.93
Command processing	126	00:00:00.65	00:00:04.14
Pass 1	96	00:00:01.08	00:00:04.57
Symbol table sort	0	00:00:00.01	00:00:00.17
Pass 2	92	00:00:00.97	00:00:04.20
Symbol table output	4	00:00:00.03	00:00:00.11
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	353	00:00:02.87	00:00:14.14

The working set limit was 900 pages.
6232 bytes (13 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 26 non-local and 7 local symbols.
479 source lines were read in Pass 1, producing 14 object records in Pass 2.
1 page of virtual memory was used to define 1 macro.

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

Macro library name

Macros defined

_S255\$DUA28:[SYSLIB]STARLET.MLB;2

0

0 GETS were required to define 0 macros.

There were no errors, warnings or information messages.

MACRO/ENABLE=SUPPRESSION/DISABLE=(GLOBAL,TRACEBACK)/LIS=LIS\$:MTHDACOS/OBJ=OBJ\$:MTHDACOS MSRC\$:MTHJACKET/UPDATE=(ENH\$:MTHJACKET)+MSRC

0258 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY