

[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

```

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

```

MT
S)

D.
D-
D-
D-
EA
FL
GE
LF
LF
LO
LS
MT
MT
MT
MT
MT
MT
MT
MT
MT
RE
SF
VA

PS
--
•
-MPh
--
In
Co
Pa
Sy
Pa
Sy
Ps
Cr
As

TH
56
TH
43
1

(2)	54	HISTORY ; Detailed Current Edit History
(3)	79	DECLARATIONS ; Declarative Part of Module
(4)	119	MTH\$ASIN - Standard Single Precision Floating ASIN
(5)	179	MTH\$ASIN_R4 - Special ASIN routine
(6)	256	MTH\$ASIND - Standard Single Precision Floating ASIND
(7)	316	MTH\$ASIND_R4 - Special ASIND routine

```
0000 1 .TITLE MTH$ASIN ; Floating Point Arcsine routine
0000 2 ; (ASIN,ASIND)
0000 3 .IDENT /1-008/ ; File: MTHASIN.MAR Edit: JCW1008
0000 4 :
0000 5 :*****
0000 6 :*
0000 7 :* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 8 :* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 9 :* ALL RIGHTS RESERVED.
0000 10 :*
0000 11 :* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 12 :* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 13 :* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 14 :* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 15 :* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 16 :* TRANSFERRED.
0000 17 :*
0000 18 :* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 19 :* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 20 :* CORPORATION.
0000 21 :*
0000 22 :* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 23 :* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 24 :*
0000 25 :*
0000 26 :*****
0000 27 :
0000 28 :
0000 29 : FACILITY: MATH LIBRARY
0000 30 :++
0000 31 : ABSTRACT:
0000 32 :
0000 33 : MTH$ASIN is a function which returns the floating point arcsine in
0000 34 : radians of its single precision floating point argument. The call is
0000 35 : standard call-by-reference.
0000 36 :
0000 37 : MTH$ASIND is a function which returns the floating point arcsine in
0000 38 : degrees of its single precision floating point argument. The call is
0000 39 : 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 MTH$ASIN
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 before
0000 64 calling MTH$$SIGNAL in order to allow user modify error result.
0000 65 1-001 - Update version number and copyright notice. JBS 16-NOV-78
0000 66 1-002 - Change MTH_INVARG to MTH$K_INVARGMAT. JBS 07-DEC-78
0000 67 1-003 - Add " " to the PSECT directive. JBS 21-DEC-78
0000 68 1-004 - Declare externals. SBL 17-May-1979
0000 69 1-005 - Use MTH$$SQRT_R3. SBL 27-Sept-1979
0000 70 1-006 - Change JSB entry to MTH$ASIN R4. RBG 28-Sept-1979
0000 71 1-007 - Added degree entry points. RNH 29-MAR-1981
0000 72 1-008 - With the aid of the RTL error plotting routine is was discovered
0000 73 that MTH$ASIN is more accurate for |X|>=0.6528662 if 1-X^2 is
0000 74 evaluated by using (1-X)*(1+X) instead of 1-X^2. Similarly
0000 75 MTH$ACOSD was found to be more accurate for |X|>=0.72929936 if
0000 76 1-X^2 was evaluated by (1-X)*(1+X).
0000 77 ;
```

```
0000 79          .SBTTL  DECLARATIONS      ; Declarative Part of Module
0000 80
0000 81 :
0000 82 : INCLUDE FILES:
0000 83 :
0000 84 :
0000 85 :
0000 86 : EXTERNAL SYMBOLS:
0000 87 :
0000 88          .DSABL  GBL
0000 89          .EXTRN  MTH$SQRT_R3
0000 90          .EXTRN  MTH$ATAN_R4
0000 91          .EXTRN  MTH$ATAN2_R4
0000 92          .EXTRN  MTH$K_INVARGMAT
0000 93          .EXTRN  MTH$$SIGNAL
0000 94
0000 95 :
0000 96 : EQUATED SYMBOLS:
0000 97
00004080 0000 98          SF_1.0 = ^F1.0          ; 1.0
OFDB40C9 0000 99          LF_PI_OV_2 = ^0007733a16 + ^0040311
0000 100          ; PI/2
000043B4 0000 101          LF_90 = ^X000043B4      ; 90
00000004 0000 102          value = 4              ; value.rf.r
0000 103
0000 104 :
0000 105 : MACROS:          none
0000 106 :
0000 107 : PSECT DECLARATIONS:
0000 108
00000000 0000 109          .PSECT  _MTH$CODE          PIC,SHR,LONG,EXE,NOWRT
0000 110          ; program section for math routines
0000 111 :
0000 112 : OWN STORAGE: none
0000 113 :
0000 114 :
0000 115 : CONSTANTS:
0000 116 :
0000 117
```



```
0000 119      .SBTTL MTH$ASIN - Standard Single Precision Floating ASIN
0000 120
0000 121
0000 122 :++
0000 123 : FUNCTIONAL DESCRIPTION:
0000 124 :
0000 125 : ASIN - single precision floating point function
0000 126 :
0000 127 : ASIN(X) is computed as:
0000 128 :
0000 129 :     If X = 0, then ASIN(X) = 0.
0000 130 :     If X = 1, then ASIN(X) = PI/2.
0000 131 :     If X = -1, then ASIN(X) = -PI/2.
0000 132 :     If 0 < |X| < 1, then ASIN(X) = ATAN(X/SQRT(1-X**2)).
0000 133 :     If 1 < |X|, error.
0000 134 :
0000 135 : CALLING SEQUENCE:
0000 136 :
0000 137 :     ASIN.wf.v = MTH$ASIN(x.rf.r)
0000 138 :
0000 139 : INPUT PARAMETERS:
0000 140 :
00000004 0000 141 :     LONG = 4 ; define longword multiplier
00000004 0000 142 :     x = 1 * LONG ; Contents of x is the argument
0000 143 :
0000 144 : IMPLICIT INPUTS: none
0000 145 :
0000 146 : OUTPUT PARAMETERS:
0000 147 :
0000 148 :     VALUE: floating arcsine of the argument
0000 149 :
0000 150 : IMPLICIT OUTPUTS: none
0000 151 :
0000 152 : COMPLETION CODES: none
0000 153 :
0000 154 : SIDE EFFECTS:
0000 155 :
0000 156 : Signals: MTH$_INVARG if |X| > 1 with reserved operand in R0 (copied to
0000 157 : the signal mechanism vector CHF$$_MCH_R0/R1 by LIB$$_SIGNAL).
0000 158 : Associated message is: "INVALID ARGUMENT". Result is reserved
0000 159 : operand -0.0 unless a user supplied (or any) error handler changes CHF$$_MCH_R0/R1
0000 160 :
0000 161 : NOTE: This procedure disables floating point underflow, enables integer
0000 162 : overflow.
0000 163 :
0000 164 : ---
0000 165 :
401C 0000 166 :
0002 167 : .ENTRY MTH$ASIN, ^M<IV, R2, R3, R4>
0002 168 : ; standard call-by-reference entry
0002 169 : ; disable DV (and FU), enable IV
0002 170 : MTH$FLAG_JACKET ; flag that this is a jacket procedure in
0002
6D 00000000'GF 9E 0002 : MOVAB G^MTH$$_JACKET_HND, (FP)
0009 : ; set handler address to jacket
0009 : handler
0009
```

```

; Floating Point Arcsine routine      M 9
MTH$ASIN - Standard Single Precision FL 16-SEP-1984 01:04:01 VAX/VMS Macro V04-00      Page 5
                                           6-SEP-1984 11:20:27 [MTHRTL.SRC]MTHASIN.MAR;1      (4)

```

```

50      04 BC      50      0009      171      MOVF      @value(AP), R0      ; case of an error in routine
      01          10      0009      172      BSBB      MTH$ASIN_R4      ; If an error, convert signal to user PC
      04          04      0009      173      RET          ; and resignal
      0010         000D      174      ; R0 = |X| = @value(AP)
      000F         000F      175      ; call special ASIN routine
      0010         0010      176      ; return with result in R0
      0010         0010      177

```



```
0010 179 .SBTTL MTH$ASIN_R4 - Special ASIN routine
0010 180
0010 181 ; Special ASIN - used by the standard routine, direct JSB call.
0010 182
0010 183 CALLING SEQUENCE:
0010 184 ; save anything needed in R0:R4
0010 185 MOVF R0 ; input in R0
0010 186 JSB MTH$ASIN_R4
0010 187 RSB ; return with result in R0
0010 188
0010 189
0010 190 MTH$ASIN_R4:: ; special ASIN routine
0010 191 MTH$ASIN_R5:: ; Release 1 name
54 50 50 0010 192 MOVF R0, R4 ; save X in R4
46 13 0013 193 BEQL RETURN ; Return ASIN(0) = 0
0015 194
0015 195 ; 0 < |X|
0015 196
0015 197
0015 198
50 8000 8F AA 0015 199 BICW #X8000, R0 ; R0 = |X|
08 50 51 001A 200 CMPF R0, S^#SF_1.0 ; compare |X| with 1.0
2C 18 001D 201 BGEQ GEQ_TO_1.0 ; branch if |X| >= 1.0
001F 202
001F 203 ; 0 < |X| < 1.0
001F 204
001F 205
001F 206
223D4027 8F 50 51 001F 207 CMPF R0, #X223D4027 ; compare |X| with 0.6528662
0C 19 0026 208 BLSS LSS_0.65 ; branch if |X| < 0.6528662
51 08 50 43 0028 209 SUBF3 R0, S^#SF_1.0, R1 ; R1 = 1 - X
50 08 40 002C 210 ADDF2 S^#SF_1.0, R0 ; R0 = 1 + X
50 51 44 002F 211 MULF2 R1, R0 ; R0 = 1.0 - X^2
07 11 0032 212 BRB FLOW ; join the normal flow
0034 213 LSS_0.65:
50 50 50 44 0034 214 MULF2 R0, R0 ; R0 = X**2
08 50 43 0037 215 SUBF3 R0, S^#SF_1.0, R0 ; R0 = 1.0 - X**2
003B 216 FLOW:
00000000'EF 16 003B 217 JSB MTH$SQRT_R3 ; R0 = SQRT(1-X**2)
50 54 50 47 0041 218 DIVF3 R0, R4, R0 ; R0 = X/SQRT(1-X**2)
00000000'EF 17 0045 219 JMP MTH$ATAN_R4 ; R0 = ATAN(X/SQRT(1-X**2))
004B 220
004B 221
004B 222 ; 1 <= |X|
004B 223
004B 224
004B 225
004B 226 GEQ_TO_1.0:
OF 14 004B 227 BGTR ERROR ; branch to ERROR if |X| > 1.0
004D 228
004D 229
004D 230 ; |X| = 1.0
004D 231
004D 232
004D 233
50 0FDB40C9 8F 50 004D 234 MOVF #LF_PI_OV_2, R0 ; R0 = PI/2
54 53 0054 235 TSTF R4 ; test the sign of X
```

MTH\$ASIN
1-008

; Floating Point Arcsine routine
MTH\$ASIN_R4 - Special ASIN routine

B 10

16-SEP-1984 01:04:01 VAX/VMS Macro V04-00
6-SEP-1984 11:20:27 [MTHRTL.SRC]MTHASIN.MAR;1

Page 7
(5)

```

50 03 18 0056 236          BGEQ  RETURN      ; branch if X > 0
50 50 52 0058 237          MNEGF  R0, R0      ; R0 = -PI/2
05 05 05B 238 RETURN: RSB                ; return with result in R0
05 05C 239
05 05C 240 ;
05 05C 241 ; 1 < |X|, error
05 05C 242 ;
05 05C 243
05 05C 244 ERROR: PUSH  (SP)                ; return PC from JSB routine
7E 00'8F 9A 005E 245 MOVZBL #MTH$K_INVARGMAT, -(SP) ; condition value
50 01 0F 78 0062 246 ASHL  #15, #T, R0      ; R0 = result = reserved operand -0.0
05 066 247 ; goes to signal mechanism vector
05 066 248 ; (CHF$L_MCH_R0/R1) so error handler
05 066 249 ; can modify the result.
00000000'GF 02 FB 0066 250 CALLS  #2, G^MTH$$SIGNAL ; signal error and use real user's PC
05 06D 251 ; independent of CALL vs JSB
05 06E 252 RSB ; return - R0 restored from CHF$L_MCH_R0/R1
05 06E 253
05 06E 254
```



```
006E 256 .SBTTL MTH$ASIND - Standard Single Precision Floating ASIND
006E 257
006E 258
006E 259 ;++
006E 260 ; FUNCTIONAL DESCRIPTION:
006E 261 ;
006E 262 ; ASIND - single precision floating point function
006E 263 ;
006E 264 ; ASIND(X) is computed as:
006E 265 ;
006E 266 ; If X = 0, then ASIND(X) = 0.
006E 267 ; If X = 1, then ASIND(X) = 90.
006E 268 ; If X = -1, then ASIND(X) = -90.
006E 269 ; If 0 < |X| < 1, then ASIND(X) = ATAND(X/SQRT(1-X**2)).
006E 270 ; If 1 < |X|, error.
006E 271 ;
006E 272 ; CALLING SEQUENCE:
006E 273 ;
006E 274 ; asind.wf.v = MTH$ASIND(x.rf.r)
006E 275 ;
006E 276 ; INPUT PARAMETERS:
006E 277 ;
00000004 006E 278 ; LONG = 4 ; define longword multiplier
00000004 006E 279 ; x = 1 * LONG ; Contents of x is the argument
006E 280 ;
006E 281 ; IMPLICIT INPUTS: none
006E 282 ;
006E 283 ; OUTPUT PARAMETERS:
006E 284 ;
006E 285 ; VALUE: floating arcsine of the argument
006E 286 ;
006E 287 ; IMPLICIT OUTPUTS: none
006E 288 ;
006E 289 ; COMPLETION CODES: none
006E 290 ;
006E 291 ; SIDE EFFECTS:
006E 292 ;
006E 293 ; Signals: MTH$_INVARG if |X| > 1 with reserved operand in R0 (copied to
006E 294 ; the signal mechanism vector CHF$_MCH_R0/R1 by LIB$SIGNAL).
006E 295 ; Associated message is: "INVALID ARGUMENT". Result is reserved
006E 296 ; operand -0.0 unless a user supplied (or any) error handler changes CHF$_MCH_R0/R1
006E 297 ;
006E 298 ; NOTE: This procedure disables floating point underflow, enables integer
006E 299 ; overflow.
006E 300 ;
006E 301 ;---
006E 302 ;
401C 006E 303 .ENTRY MTH$ASIND, ^M<IV, R2, R3, R4>
0070 304 ; standard call-by-reference entry
0070 305 ; disable DV (and FU), enable IV
0070 306 MTH$FLAG_JACKET ; flag that this is a jacket procedure in
0070 307
6D 00000000'GF 9E 0070 MOVAB G^MTH$$JACKET_HND, (FP)
0077 ; set handler address to jacket
0077 ; handler
0077
```

```

; Floating Point Arcsine routine      D 10
MTH$ASIND - Standard Single Precision F 16-SEP-1984 01:04:01 VAX/VMS Macro V04-00      Page 9
                                           6-SEP-1984 11:20:27 [MTHRTL.SRC]MTHASIN.MAR;1      (6)

```

			0077	308			: case of an error in routine
			0077	309			: If an error, convert signal to user PC
			0077	310			: and resignal
50	04	BC	50	0077	311	MOVF @value(AP), R0	: R0 = X = @value(AP)
		01	10	007B	312	BSBB MTH\$ASIND_R4	: call special ASIND routine
			04	007D	313	RET	: return with result in R0
			007E	314			


```
007E 316 .SBTTL MTH$ASIND_R4 - Special ASIND routine
007E 317
007E 318 ; Special ASIND - used by the standard routine, direct JSB call.
007E 319
007E 320 CALLING SEQUENCE:
007E 321 ; save anything needed in R0:R4
007E 322 MOVF R0 ; input in R0
007E 323 JSB MTH$ASIND_R4
007E 324 RSB ; return with result in R0
007E 325
007E 326
007E 327 MTH$ASIND_R4:: ; special ASIND routine
54 50 50 007E 328 MOVF R0, R4 ; save X in R4
46 13 0081 329 BEQL D_RETURN ; Return ASIND(0) = 0
0083 330
0083 331 ; 0 < !X!
0083 332
0083 333
0083 334
50 8000 8F AA 0083 335 BICW #^X8000, R0 ; R0 = !X!
08 50 51 0088 336 CMPF R0, S^#SF_1.0 ; compare !X! with 1.0
2C 18 008B 337 BGEQ D_GEQ_TO_1.0 ; branch if !X! >= 1.0
008D 338
008D 339 ; 0 < !X! < 1.0
008D 340
008D 341
008D 342
B35C403A 8F 50 51 008D 343 CMPF R0, #^XB35C403A ; compare !X! with 0.72929936
0C 19 0094 344 BLSS D_LSS_0.729 ; branch if !X! < 0.72929936
51 08 50 43 0096 345 SUBF3 R0, S^#SF_1.0, R1 ; R1 = 1 - X
50 08 40 009A 346 ADDF2 S^#SF_1.0, R0 ; R0 = 1 + X
50 51 44 009D 347 MULF2 R1, R0 ; R0 = 1.0 - X^2
07 11 00A0 348 BRB D_FLOW ; join the normal flow
00A2 349 D_LSS_0.729:
50 50 50 44 00A2 350 MULF2 R0, R0 ; R0 = X**2
08 50 43 00A5 351 SUBF3 R0, S^#SF_1.0, R0 ; R0 = 1.0 - X**2
00A9 352 D_FLOW:
00000000'EF 16 00A9 353 JSB MTH$SQRT_R3 ; R0 = SQRT(1-X**2)
50 54 50 47 00AF 354 DIVF3 R0, R4, R0 ; R0 = X/SQRT(1-X**2)
00000000'EF 17 00B3 355 JMP MTH$ATAND_R4 ; R0 = ATAND(X/SQRT(1-X**2))
00B9 356
00B9 357
00B9 358 ; 1 =< !X!
00B9 359
00B9 360
00B9 361
00B9 362 D_GEQ_TO_1.0:
A1 14 00B9 363 BGTR ERROR ; branch to ERROR if !X! > 1.0
00BB 364
00BB 365
00BB 366 ; !X! = 1.0
00BB 367
00BB 368
00BB 369
50 000043B4 8F 50 00BB 370 MOVF #LF_90, R0 ; R0 = 90
54 53 00C2 371 TSTF R4 ; test the sign of X
95 18 00C4 372 BGEQ RETURN ; branch if X > 0
```

MTH\$ASIN
1-008

F 10
; Floating Point Arcsine routine
MTH\$ASIND_R4 - Special ASIND routine

16-SEP-1984 01:04:01 VAX/VMS Macro V04-00
6-SEP-1984 11:20:27 [MTHRTL.SRC]MTHASIN.MAR;1

Page 11
(7)

```
50 50 52 00C6 373 MNEGF R0, R0
      00C9 374 D_RETURN:
      05 00C9 375 RSB
      00CA 376
      00CA 377
      00CA 378
      00CA 379 .END
```

```
; R0 = -90
; return with result in R0
```


MTH\$ASIN
Symbol table

; Floating Point Arcsine routine G 10

16-SEP-1984 01:04:01 VAX/VMS Macro V04-00
6-SEP-1984 11:20:27 [MTHRTL.SRC]MTHASIN.MAR;1

Page 12
(7)

D_FLOW	000000A9	R	01
D_GEQ_TO_1.0	000000B9	R	01
D_LSS_0.729	000000A2	R	01
D_RETURN	000000C9	R	01
ERROR	0000005C	R	01
FLOW	0000003B	R	01
GEQ_TO_1.0	0000004B	R	01
LF_90	= 000043B4		
LF_PI_OV_2	= 0FDB40C9		
LONG	= 00000004		
LSS_0.65	00000034	R	01
MTH\$\$JACKET_HND	*****	X	01
MTH\$\$SIGNAL	*****	X	00
MTH\$ASIN	00000000	RG	01
MTH\$ASIND	0000006E	RG	01
MTH\$ASIND_R4	0000007E	RG	01
MTH\$ASIN_R4	00000010	RG	01
MTH\$ASIN_R5	00000010	RG	01
MTH\$ATAN_R4	*****	X	00
MTH\$ATAN_R4	*****	X	00
MTH\$K_INVARGMAT	*****	X	00
MTH\$SQRT_R3	*****	X	00
RETURN	0000005B	R	01
SF_1.0	= 00004080		
VALUE	= 00000004		

! Psect synopsis !

PSECT name	Allocation	PSECT No.	Attributes												
ABS	00000000 (0.)	00 (0.)	NOPI	USR	CON	ABS	LCL	NOSHR	NOEXE	NORD	NOWRT	NOVEC	BYTE		
_MTH\$CODE	000000CA (202.)	01 (1.)	PIC	USR	CON	REL	LCL	SHR	EXE	RD	NOWRT	NOVEC	LONG		

! Performance indicators !

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.09	00:00:00.74
Command processing	122	00:00:00.77	00:00:03.89
Pass 1	95	00:00:00.97	00:00:03.61
Symbol table sort	0	00:00:00.00	00:00:00.07
Pass 2	77	00:00:00.88	00:00:02.62
Symbol table output	4	00:00:00.04	00:00:00.12
Psect synopsis output	2	00:00:00.03	00:00:00.05
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	331	00:00:02.78	00:00:11.19

The working set limit was 750 pages.
5683 bytes (12 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 0 local symbols.
439 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\$:MTHASIN/OBJ=OBJ\$:MTHASIN MSRC\$:MTHJACKET/UPDATE=(ENH\$:MTHJACKET)+MSRC\$:

0257 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

