


```
FFFFFFFFF 000000 RRRRRRRR VV VV EEEEEEEEE CCCCCCCC TTTTTTTTT 000000 RRRRRRRR
FFFFFFFFF 000000 RRRRRRRR VV VV EEEEEEEEE CCCCCCCC TTTTTTTTT 000000 RRRRRRRR
FF          00      00 RR      RR VV      VV EE          CC          TT          00      00 RR      RR
FF          00      00 RR      RR VV      VV EE          CC          TT          00      00 RR      RR
FF          00      00 RR      RR VV      VV EE          CC          TT          00      00 RR      RR
FF          00      00 RR      RR VV      VV EE          CC          TT          00      00 RR      RR
FFFFFFFFF 00      00 RRRRRRRR VV      VV EEEEEEEE CC          TT          00      00 RRRRRRRR
FFFFFFFFF 00      00 RRRRRRRR VV      VV EEEEEEEE CC          TT          00      00 RRRRRRRR
FF          00      00 RR      RR VV      VV EE          CC          TT          00      00 RR      RR
FF          00      00 RR      RR VV      VV EE          CC          TT          00      00 RR      RR
FF          00      00 RR      RR VV      VV EE          CC          TT          00      00 RR      RR
FF          00      00 RR      RR VV      VV EE          CC          TT          00      00 RR      RR
FF          00      00 RR      RR VV      VV EE          CC          TT          00      00 RR      RR
FF          000000 RR      RR VV      VV EEEEEEEEE CCCCCCCC TTT          000000 RR      RR
FF          000000 RR      RR VV      VV EEEEEEEEE CCCCCCCC TTT          000000 RR      RR
```

```
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
LLLLLLLLLLL IIIIII SSSSSSSS
LLLLLLLLLLL IIIIII SSSSSSSS
```


FOR\$VECTOR
Table of contents

- Entry vectors for FORRTL.EXE

F 4

15-SEP-1984 23:46:45 VAX/VMS Macro V04-00

Page 0

(2) 50
(3) 113

DECLARATIONS
FORRTL Vector

F
V

M
I
O
T
M

```
0000 1 .TITLE FOR$VECTOR - Entry vectors for FORRTL.EXE
0000 2 .IDENT /1-004/ ; File: FORVECTOR.MAR Edit: SBL1004
0000 3
0000 4
0000 5 *****
0000 6
0000 7 *
0000 8 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 9 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 10 * ALL RIGHTS RESERVED.
0000 11 *
0000 12 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 13 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 14 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 15 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 16 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 17 * TRANSFERRED.
0000 18 *
0000 19 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 20 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 21 * CORPORATION.
0000 22 *
0000 23 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 24 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 25 *
0000 26 *****
0000 27
0000 28
0000 29 ++
0000 30 : FACILITY: Run-Time Library - FORTRAN Language Support
0000 31 :
0000 32 : ABSTRACT:
0000 33 :
0000 34 : This module contains the entry vector definitions for the
0000 35 : VAX-11 Run-Time Library shareable image FORRTL.EXE
0000 36 :
0000 37 : ENVIRONMENT: User mode, AST Reentrant
0000 38 :
0000 39 : AUTHOR: Steven B. Lionel, CREATION DATE: 29-October-1982
0000 40 :
0000 41 : MODIFIED BY:
0000 42 :
0000 43 : 1-001 - Original. SBL 29-October-1982
0000 44 : 1-002 - Add FOR$$IO BEG. SBL 11-May-1983
0000 45 : 1-003 - Add FOR$READ_IL and FOR$WRITE_IL. SBL 19-May-1983
0000 46 : 1-004 - Add all remaining FOR$ routines, plus those FOR$$ routines
0000 47 : referenced by COM$ routines. SBL 1-Jul-1983
0000 48 :--
```



```
0000 50      .SBTTL  DECLARATIONS
0000 51      :
0000 52      : LIBRARY MACRO CALLS:
0000 53      :
0000 54      :     NONE
0000 55      :
0000 56      : EXTERNAL DECLARATIONS:
0000 57      :
0000 58      :     .DSABL  GBL                ; Force all external symbols to be declared
0000 59      :
0000 60      : MACROS:
0000 61      :
0000 62      :
0000 63      :+
0000 64      : Macro to define an entry vector for a CALL entry point
0000 65      : -
0000 66      :
0000 67      : .MACRO  VCALL      NAME, ALTMSK
0000 68      : .EXTRN      NAME
0000 69      : .TRANSFER  NAME
0000 70      : .IF B ALTMSK
0000 71      : .MASK      NAME
0000 72      : .IFF
0000 73      : .MASK      ALTMSK
0000 74      : .ENDC
0000 75      : JMP      NAME+2
0000 76      : .ENDM
0000 77      :
0000 78      :+
0000 79      : Macro to define an entry vector for a JSB entry point
0000 80      : -
0000 81      :
0000 82      : .MACRO  VJSB      NAME
0000 83      : .EXTRN      NAME
0000 84      : .TRANSFER  NAME
0000 85      : JMP      NAME
0000 86      : .BLKB      2
0000 87      : .ENDM
0000 88      :
0000 89      :+
0000 90      : Macro to define an alias for the next vectored entry point
0000 91      : -
0000 92      :
0000 93      : .MACRO  ALIAS      NAME
0000 94      : .TRANSFER  NAME
0000 95      : .ENDM
0000 96      :
0000 97      :
0000 98      :
0000 99      : EQUATED SYMBOLS:
0000 100     :
0000 101     :     NONE
0000 102     :
0000 103     : OWN STORAGE:
0000 104     :
0000 105     :     NONE
0000 106     :
```

FOR\$VECTOR
1-004

- Entry vectors for FORRTL.EXE I 4
DECLARATIONS

15-SEP-1984 23:46:45 VAX/VMS Macro V04-00 Page 3
6-SEP-1984 11:01:42 [FORRTL.SRC]FORVECTOR.MAR;1 (2)

0000	107	:	PSECT DECLARATIONS:
0000	108	:	
00000000	109	:	
0000	110	:	
0000	111	:	

.PSECT \$FOR\$VECTOR PIC, USR, CON, REL, LCL, SHR, -
EXE, RD, NOWRT, LONG


```
0000 113      .SBTTL  FORRTL Vector
0000 114
0000 115      ;+
0000 116      ; Define vectored entry points for the FORTRAN Language Support procedures
0000 117      ; by module in alphabetical order.
0000 118
0000 119      ; Any additions to this file should be reflected in
0000 120      ; COM$;FORRTLVEC.DAT. All new entry points must be appended to the end
0000 121      ; of the list. NEVER change existing entries unless you are sure that
0000 122      ; what you do won't break existing programs.
0000 123      ; -
0000 124
0000 125      ; Module FOR$$CB
0000 126
0000 127          VJSB  FOR$$CB_GET
0008 128          ALIAS FOR$$CB_RET
0008 129          VJSB  FOR$$CB_POP
0010 130          VJSB  FOR$$CB_PUSH
0018 131          VCALL FOR$$FP_MATCH
0020 132
0020 133      ; Module FOR$BACKSPACE
0020 134
0020 135          VCALL  FOR$BACKSPACE
0028 136
0028 137      ; Module FOR$CLOSE
0028 138
0028 139          VCALL  FOR$CLOSE
0030 140
0030 141      ; Module FOR$CVTRT
0030 142
0030 143          ALIAS  FOR$CNV_OUT_D
0030 144          VCALL  FOR$CVT_D_TD
0038 145          ALIAS  FOR$CNV_OUT_E
0038 146          VCALL  FOR$CVT_D_TE
0040 147          ALIAS  FOR$CNV_OUT_F
0040 148          VCALL  FOR$CVT_D_TF
0048 149          ALIAS  FOR$CNV_OUT_G
0048 150          VCALL  FOR$CVT_D_TG
0050 151          VCALL  FOR$CVT_F_TD
0058 152          VCALL  FOR$CVT_F_TE
0060 153          VCALL  FOR$CVT_F_TF
0068 154          VCALL  FOR$CVT_F_TG
0070 155          VCALL  FOR$CVT_G_TD
0078 156          VCALL  FOR$CVT_G_TE
0080 157          VCALL  FOR$CVT_G_TF
0088 158          VCALL  FOR$CVT_G_TG
0090 159          VCALL  FOR$CVT_H_TD
0098 160          VCALL  FOR$CVT_H_TE
00A0 161          VCALL  FOR$CVT_H_TF
00A8 162          VCALL  FOR$CVT_H_TG
00B0 163
00B0 164      ; Module FOR$DECODE_MF
00B0 165
00B0 166          VCALL  FOR$DECODE_MF  FOR$$IO_BEG
00B8 167
00B8 168      ; Module FOR$DECODE_MO
00B8 169
```

```
00B8 170          VCALL  FOR$DECODE_MO  FOR$$IO_BEG
00C0 171
00C0 172 ; Module FOR$DEFINE_FILE
00C0 173
00C0 174          VCALL  FOR$DEF_FILE
00C8 175          VCALL  FOR$DEF_FILE_W
00D0 176
00D0 177 ; Module FOR$DELETE
00D0 178
00D0 179          VCALL  FOR$DELETE
00D8 180          VCALL  FOR$DELETE_D
00E0 181
00E0 182 ; Module FOR$ENCODE_MF
00E0 183
00E0 184          VCALL  FOR$ENCODE_MF  FOR$$IO_BEG
00E8 185
00E8 186 ; Module FOR$ENCODE_MO
00E8 187
00E8 188          VCALL  FOR$ENCODE_MO  FOR$$IO_BEG
00F0 189
00F0 190 ; Module FOR$ENDFILE
00F0 191
00F0 192          VCALL  FOR$ENDFILE
00F8 193
00F8 194 ; Module FOR$ERRSNS
00F8 195
00F8 196          VCALL  FOR$$ERRSNS_SAV
0100 197          VCALL  FOR$ERRSNS
0108 198          VCALL  FOR$ERRSNS_W
0110 199
0110 200 ; Module FOR$EXIT
0110 201
0110 202          VCALL  FOR$EXIT
0118 203          VCALL  FOR$EXIT_W
0120 204
0120 205 ; Module FOR$FIND
0120 206
0120 207          VCALL  FOR$FIND
0128 208
0128 209 ; Module FOR$INI_DES
0128 210
0128 211          VJSB   FOR$INI_DES1_R2
0130 212          VJSB   FOR$INI_DES2_R3
0138 213          VJSB   FOR$INI_DESC_R6
0140 214
0140 215 ; Module FOR$INQUIRE
0140 216
0140 217          VCALL  FOR$INQUIRE
0148 218
0148 219 ; Module FOR$IO_ELEM
0148 220
0148 221          VCALL  FOR$IO_B_R
0150 222          VCALL  FOR$IO_B_V
0158 223          VCALL  FOR$IO_DC_R
0160 224          VCALL  FOR$IO_DC_V
0168 225          VCALL  FOR$IO_D_R
0170 226          VCALL  FOR$IO_D_V
```


0178	227	VCALL	FOR\$IO_FC_R	
0180	228	VCALL	FOR\$IO_FC_V	
0188	229	VCALL	FOR\$IO_F_R	
0190	230	VCALL	FOR\$IO_F_V	
0198	231	VCALL	FOR\$IO_GC_R	
01A0	232	VCALL	FOR\$IO_GC_V	
01A8	233	VCALL	FOR\$IO_G_R	
01B0	234	VCALL	FOR\$IO_G_V	
01B8	235	VCALL	FOR\$IO_H_R	
01C0	236	VCALL	FOR\$IO_H_V	
01C8	237	VCALL	FOR\$IO_LO_R	
01D0	238	VCALL	FOR\$IO_LU_V	
01D8	239	VCALL	FOR\$IO_L_R	
01E0	240	VCALL	FOR\$IO_L_V	
01E8	241	VCALL	FOR\$IO_T_DS	
01F0	242	VCALL	FOR\$IO_T_V_DS	
01F8	243	VCALL	FOR\$IO_WO_R	
0200	244	VCALL	FOR\$IO_WU_V	
0208	245	VCALL	FOR\$IO_W_R	
0210	246	VCALL	FOR\$IO_W_V	
0218	247	VCALL	FOR\$IO_X_DA	
0220	248	VCALL	FOR\$IO_X_NL	
0228	249	VCALL	FOR\$IO_X_SB	
0230	250	VCALL	FOR\$IO_X_SE	
0238	251			
0238	252	; Module FOR\$IO_END		
0238	253			
0238	254	VCALL	FOR\$IO_END	
0240	255			
0240	256	; Module FOR\$OPEN		
0240	257			
0240	258	VCALL	FOR\$OPEN	
0248	259			
0248	260	; Module FOR\$PAUSE		
0248	261			
0248	262	VCALL	FOR\$PAUSE	
0250	263			
0250	264	; Module FOR\$RAB		
0250	265			
0250	266	VCALL	FOR\$RAB	
0258	267			
0258	268	; Module FOR\$READ_DF		
0258	269			
0258	270	VCALL	FOR\$READ_DF	FOR\$\$IO_BEG
0260	271			
0260	272	; Module FOR\$READ_DO		
0260	273			
0260	274	VCALL	FOR\$READ_DO	FOR\$\$IO_BEG
0268	275			
0268	276	; Module FOR\$READ_DU		
0268	277			
0268	278	VCALL	FOR\$READ_DU	FOR\$\$IO_BEG
0270	279			
0270	280	; Module FOR\$READ_IF		
0270	281			
0270	282	VCALL	FOR\$READ_IF	FOR\$\$IO_BEG
0278	283			

0278	284	: Module FOR\$READ_IO	
0278	285		
0278	286	VCALL FOR\$READ_IO	FOR\$\$IO_BEG
0280	287		
0280	288	: Module FOR\$READ_KF	
0280	289		
0280	290	VCALL FOR\$READ_KF	FOR\$\$IO_BEG
0288	291		
0288	292	: Module FOR\$READ_KO	
0288	293		
0288	294	VCALL FOR\$READ_KO	FOR\$\$IO_BEG
0290	295		
0290	296	: Module FOR\$READ_KU	
0290	297		
0290	298	VCALL FOR\$READ_KU	FOR\$\$IO_BEG
0298	299		
0298	300	: Module FOR\$READ_SF	
0298	301		
0298	302	VCALL FOR\$READ_SF	FOR\$\$IO_BEG
02A0	303		
02A0	304	: Module FOR\$READ_SL	
02A0	305		
02A0	306	VCALL FOR\$READ_SL	FOR\$\$IO_BEG
02A8	307		
02A8	308	: Module FOR\$READ_SN	
02A8	309		
02A8	310	VCALL FOR\$READ_SN	FOR\$\$IO_BEG
02B0	311		
02B0	312	: Module FOR\$READ_SO	
02B0	313		
02B0	314	VCALL FOR\$READ_SO	FOR\$\$IO_BEG
02B8	315		
02B8	316	: Module FOR\$READ_SU	
02B8	317		
02B8	318	VCALL FOR\$READ_SU	FOR\$\$IO_BEG
02C0	319		
02C0	320	: Module FOR\$REWIND	
02C0	321		
02C0	322	VCALL FOR\$REWIND	
02C8	323		
02C8	324	: Module FOR\$REWRITE_SF	
02C8	325		
02C8	326	VCALL FOR\$REWRITE_SF	FOR\$\$IO_BEG
02D0	327		
02D0	328	: Module FOR\$REWRITE_SO	
02D0	329		
02D0	330	VCALL FOR\$REWRITE_SO	FOR\$\$IO_BEG
02D8	331		
02D8	332	: Module FOR\$REWRITE_SU	
02D8	333		
02D8	334	VCALL FOR\$REWRITE_SU	FOR\$\$IO_BEG
02E0	335		
02E0	336	: Module FOR\$SECNDS	
02E0	337		
02E0	338	VCALL FOR\$SECNDS	
02E8	339		
02E8	340	: Module FOR\$STOP	


```
02E8 341
02E8 342          VCALL  FOR$STOP
02F0 343
02F0 344 ; Module FOR$UNLOCK
02F0 345
02F0 346          VCALL  FOR$UNLOCK
02F8 347
02F8 348 ; Module FOR$WRITE_DF
02F8 349
02F8 350          VCALL  FOR$WRITE_DF  FOR$$IO_BEG
0300 351
0300 352 ; Module FOR$WRITE_DO
0300 353
0300 354          VCALL  FOR$WRITE_DO  FOR$$IO_BEG
0308 355
0308 356 ; Module FOR$WRITE_DU
0308 357
0308 358          VCALL  FOR$WRITE_DU  FOR$$IO_BEG
0310 359
0310 360 ; Module FOR$WRITE_IF
0310 361
0310 362          VCALL  FOR$WRITE_IF  FOR$$IO_BEG
0318 363
0318 364 ; Module FOR$WRITE_IO
0318 365
0318 366          VCALL  FOR$WRITE_IO  FOR$$IO_BEG
0320 367
0320 368 ; Module FOR$WRITE_SF
0320 369
0320 370          VCALL  FOR$WRITE_SF  FOR$$IO_BEG
0328 371
0328 372 ; Module FOR$WRITE_SL
0328 373
0328 374          VCALL  FOR$WRITE_SL  FOR$$IO_BEG
0330 375
0330 376 ; Module FOR$WRITE_SN
0330 377
0330 378          VCALL  FOR$WRITE_SN  FOR$$IO_BEG
0338 379
0338 380 ; Module FOR$WRITE_SO
0338 381
0338 382          VCALL  FOR$WRITE_SO  FOR$$IO_BEG
0340 383
0340 384 ; Module FOR$WRITE_SU
0340 385
0340 386          VCALL  FOR$WRITE_SU  FOR$$IO_BEG
0348 387
0348 388
0348 389 ;+
0348 390 ; End of initial FORRTL vector. All subsequent additions must be made
0348 391 ; after this point.
0348 392 ; -
0348 393
0348 394 ;+
0348 395 ; FOR$$IO_BEG is never called from outside of FORRTL, but its entry mask
0348 396 ; needs to be made available to VMSRTL, hence its inclusion here.
0348 397 ; -
```



```
0348 398
0348 399 ; Module FOR$$IO_BEG
0348 400         VCALL  FOR$$IO_BEG
0350 401
0350 402 ; Module FOR$READ_IL
0350 403         VCALL  FOR$READ_IL      FOR$$IO_BEG
0358 404
0358 405 ; Module FOR$WRITE_IL
0358 406         VCALL  FOR$WRITE_IL     FOR$$IO_BEG
0360 407
0360 408 ; Module FOR$BITOPS
0360 409
0360 410         VCALL  FOR$BITEST
0368 411         VCALL  FOR$BJTEST
0370 412         VCALL  FOR$IIBCLR
0378 413         VCALL  FOR$IIBITS
0380 414         VCALL  FOR$IIBSET
0388 415         VCALL  FOR$IISHFTC
0390 416         VCALL  FOR$IMVBITS
0398 417         VCALL  FOR$JIBCLR
03A0 418         VCALL  FOR$JIBITS
03A8 419         VCALL  FOR$JIBSET
03B0 420         VCALL  FOR$JISHFTC
03B8 421         VCALL  FOR$JMVBITS
03C0 422
03C0 423 ; Module FOR$DATE
03C0 424
03C0 425         VCALL  FOR$DATE
03C8 426
03C8 427 ; Module FOR$DATE_T_DS
03C8 428
03C8 429         VCALL  FOR$DATE_T_DS
03D0 430
03D0 431 ; Module FOR$IDATE
03D0 432
03D0 433         VCALL  FOR$IDATE
03D8 434
03D8 435 ; Module FOR$JDATE
03D8 436
03D8 437         VCALL  FOR$JDATE
03E0 438
03E0 439 ; Module FOR$LEX
03E0 440
03E0 441         VCALL  FOR$LGE
03E8 442         VCALL  FOR$LGT
03F0 443         VCALL  FOR$LLE
03F8 444         VCALL  FOR$LLT
0400 445
0400 446 ; Module FOR$RANDOM
0400 447
0400 448         VCALL  FOR$IRAN
0408 449         ALIAS  FOR$RANDU_W
0408 450         VCALL  FOR$RANDU
0410 451
0410 452 ; Module FOR$TIME
0410 453
0410 454         VCALL  FOR$TIME
```


FOR\$VECTOR
1-004

- Entry vectors for FORRTL.EXE
FORRTL Vector

C 5

15-SEP-1984 23:46:45
6-SEP-1984 11:01:42

VAX/VMS Macro V04-00
[FORRTL.SRC]FORVECTOR.MAR;1

Page 10
(3)

```
0418 455
0418 456 ; Module FOR$TIME_T_DS
0418 457
0418 458          VCALL  FOR$TIME_T_DS
0420 459
0420 460 ;+
0420 461 ; Internal entry points that are called by COM$ (PDP-11 Compatibility)
0420 462 ; modules.
0420 463 ; -
0420 464
0420 465 ; Module FOR$$ERROR
0420 466
0420 467          VCALL  FOR$$ERR_OPECLO
0428 468
0428 469 ; Module FOR$$SIGNAL
0428 470
0428 471          VCALL  FOR$$SIGNAL_STO
0430 472          VCALL  FOR$$SIG_FATINT
0438 473          VCALL  FOR$$SIG_NO_LUB
0440 474
0440 475 ; Module FOR$ERRSNS
0440 476
0440 477          VCALL  FOR$$INIT_ERRSET
0448 478
0448 479 ; Module FOR$$VM
0448 480
0448 481          VCALL  FOR$$FREE_VM
0450 482          VCALL  FOR$$GET_VM
0458 483
0458 484          .END
```

; End of module FOR\$VECTOR

Page 11
(3)

FOR\$	NAME	TYPE	MODE	PERM	STATUS
FOR\$SCB	GET				X
FOR\$SCB	POP				X
FOR\$SCB	PUSH				X
FOR\$ERRS	SNS SAV				X
FOR\$ERR	OPECLO				X
FOR\$FP	MATCH				X
FOR\$FREE	VM				X
FOR\$GET	VM				X
FOR\$INIT	ERRSET				X
FOR\$IO	BEG				X
FOR\$SIGNAL	STO				X
FOR\$SIG	FATINT				X
FOR\$SIG	NO LUB				X
FOR\$BACKSPACE					X
FOR\$BITEST					X
FOR\$BJTEST					X
FOR\$CLOSE					X
FOR\$CVT	D TD				X
FOR\$CVT	D TE				X
FOR\$CVT	D TF				X
FOR\$CVT	D TG				X
FOR\$CVT	F TD				X
FOR\$CVT	F TE				X
FOR\$CVT	F TF				X
FOR\$CVT	F TG				X
FOR\$CVT	G TD				X
FOR\$CVT	G TE				X
FOR\$CVT	G TF				X
FOR\$CVT	G TG				X
FOR\$CVT	H TD				X
FOR\$CVT	H TE				X
FOR\$CVT	H TF				X
FOR\$CVT	H TG				X
FOR\$DATE					X
FOR\$DATE	T_DS				X
FOR\$DECODE	MF				X
FOR\$DECODE	MO				X
FOR\$DEF	FILE				X
FOR\$DEF	FILE_W				X
FOR\$DELETE					X
FOR\$DELETE	D				X
FOR\$ENCODE	MF				X
FOR\$ENCODE	MO				X
FOR\$ENDFILE					X
FOR\$ERRS					X
FOR\$ERRS	_W				X
FOR\$EXIT					X
FOR\$EXIT	_W				X
FOR\$FIND					X
FOR\$IDATE					X
FOR\$IIBCLR					X
FOR\$IIBITS					X
FOR\$IIBSET					X
FOR\$IISHFTC					X
FOR\$IMVBITS					X
FOR\$INI	DES1_R2				X
FOR\$INI	DES2_R3				X

FOR\$INI DESC_R6
FOR\$INQUIRE
FOR\$IO_B_R
FOR\$IO_B_V
FOR\$IO_DC_R
FOR\$IO_DC_V
FOR\$IO_D_R
FOR\$IO_D_V
FOR\$IO_END
FOR\$IO_FC_R
FOR\$IO_FC_V
FOR\$IO_F_R
FOR\$IO_F_V
FOR\$IO_GC_R
FOR\$IO_GC_V
FOR\$IO_G_R
FOR\$IO_G_V
FOR\$IO_H_R
FOR\$IO_H_V
FOR\$IO_LO_R
FOR\$IO_LU_V
FOR\$IO_L_R
FOR\$IO_L_V
FOR\$IO_T_DS
FOR\$IO_T_V_DS
FOR\$IO_WO_DS
FOR\$IO_WU_V
FOR\$IO_W_R
FOR\$IO_W_V
FOR\$IO_X_DA
FOR\$IO_X_NL
FOR\$IO_X_SB
FOR\$IO_X_SE
FOR\$IRAN
FOR\$JDATE
FOR\$JIBCLR
FOR\$JIBITS
FOR\$JIBSET
FOR\$JISHFTC
FOR\$JMVBITS
FOR\$LGE
FOR\$LGT
FOR\$LLE
FOR\$LLT
FOR\$OPEN
FOR\$PAUSE
FOR\$RAB
FOR\$RANDU
FOR\$READ_DF
FOR\$READ_DO
FOR\$READ_DU
FOR\$READ_IF
FOR\$READ_IL
FOR\$READ_IO
FOR\$READ_KF
FOR\$READ_KO
FOR\$READ_KU

[illegible]

FOR\$VECTOR
Symbol table

- Entry vectors for FORRTL.EXE

E 5

15-SEP-1984 23:46:45
6-SEP-1984 11:01:42

VAX/VMS Macro V04-00
[FORRTL.SRC]FORVECTOR.MAR;1

Page 12
(3)

FOR\$READ_SF	*****	X	01
FOR\$READ_SL	*****	X	01
FOR\$READ_SN	*****	X	01
FOR\$READ_SO	*****	X	01
FOR\$READ_SU	*****	X	01
FOR\$REWIND	*****	X	01
FOR\$REWRITE_SF	*****	X	01
FOR\$REWRITE_SO	*****	X	01
FOR\$REWRITE_SU	*****	X	01
FOR\$SECNDS	*****	X	01
FOR\$STOP	*****	X	01
FOR\$TIME	*****	X	01
FOR\$TIME_T_DS	*****	X	01
FOR\$UNLOCK	*****	X	01
FOR\$WRITE_DF	*****	X	01
FOR\$WRITE_DO	*****	X	01
FOR\$WRITE_DU	*****	X	01
FOR\$WRITE_IF	*****	X	01
FOR\$WRITE_IL	*****	X	01
FOR\$WRITE_IO	*****	X	01
FOR\$WRITE_SF	*****	X	01
FOR\$WRITE_SL	*****	X	01
FOR\$WRITE_SN	*****	X	01
FOR\$WRITE_SO	*****	X	01
FOR\$WRITE_SU	*****	X	01

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

PSECT name	Allocation	PSECT No.	Attributes												
ABS	00000000 (0.)	00 (0.)	NOPIC	USR	CON	ABS	LCL	NOSHR	NOEXE	NORD	NOWRT	NOVEC	BYTE		
\$FOR\$VECTOR	00000458 (1112.)	01 (1.)	PIC	USR	CON	REL	LCL	SHR	EXE	RD	NOWRT	NOVEC	LONG		

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

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.12	00:00:00.95
Command processing	114	00:00:00.54	00:00:07.35
Pass 1	122	00:00:03.30	00:00:08.16
Symbol table sort	0	00:00:00.15	00:00:00.19
Pass 2	102	00:00:01.46	00:00:03.72
Symbol table output	18	00:00:00.13	00:00:00.66
Psect synopsis output	1	00:00:00.02	00:00:00.05
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	388	00:00:05.73	00:00:21.08

The working set limit was 1050 pages.
17467 bytes (35 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 139 non-local and 0 local symbols.
484 source lines were read in Pass 1, producing 29 object records in Pass 2.
3 pages of virtual memory were used to define 3 macros.

+-----+
! 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/LIS=LIS\$:FORVECTOR/OBJ=OBJ\$:FORVECTOR MSRC\$:FORVECTOR/UPDATE=(ENH\$:FORVECTOR)

0185

AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY