```
EEEEEEEEEEEEEEE  RRRRRRRRRRRR     FFFFFFFFFFFFFFF
EEEEEEEEEEEEEEE  RRRRRRRRRRRR     FFFFFFFFFFFFFFF
EEEEEEEEEEEEEEE  RRRRRRRRRRRR     FFFFFFFFFFFFFFF
EEE              RRR         RRR  FFF
EEE              RRR         RRR  FFF
EEE              RRR         RRR  FFF
EEE              RRR         RRR  FFF
EEE              RRR         RRR  FFF
EEE              RRR         RRR  FFF
EEEEEEEEEEEE     RRRRRRRRRRRR     FFFFFFFFFFFF
EEEEEEEEEEEE     RRRRRRRRRRRR     FFFFFFFFFFFF
EEEEEEEEEEEE     RRRRRRRRRRRR     FFFFFFFFFFFF
EEE              RRR   RRR        FFF
EEE              RRR   RRR        FFF
EEE              RRR   RRR        FFF
EEE              RRR      RRR     FFF
EEE              RRR      RRR     FFF
EEE              RRR      RRR     FFF
EEEEEEEEEEEEEEE  RRR         RRR  FFF
EEEEEEEEEEEEEEE  RRR         RRR  FFF
EEEEEEEEEEEEEEE  RRR         RRR  FFF
```

FILEID**DHEADS

```
DDDDDDDD    HH       HH  EEEEEEEEEE     AAAAAA     DDDDDDDD        SSSSSSSS
DDDDDDDD    HH       HH  EEEEEEEEEE     AAAAAA     DDDDDDDD        SSSSSSSS
DD      DD  HH       HH  EE           AA      AA   DD      DD    SS
DD      DD  HH       HH  EE           AA      AA   DD      DD    SS
DD      DD  HH       HH  EE           AA      AA   DD      DD    SS
DD      DD  HH       HH  EE           AA      AA   DD      DD    SS
DD      DD  HHHHHHHHHHH  EEEEEEEE     AA      AA   DD      DD      SSSSSS
DD      DD  HHHHHHHHHHH  EEEEEEEE     AA      AA   DD      DD      SSSSSS
DD      DD  HH       HH  EE           AAAAAAAAAA   DD      DD            SS
DD      DD  HH       HH  EE           AAAAAAAAAA   DD      DD            SS
DD      DD  HH       HH  EE           AA      AA   DD      DD            SS    ....
DD      DD  HH       HH  EE           AA      AA   DD      DD            SS    ....
DDDDDDDD    HH       HH  EEEEEEEEEE   AA      AA   DDDDDDDD      SSSSSSSS      ....
DDDDDDDD    HH       HH  EEEEEEEEEE   AA      AA   DDDDDDDD      SSSSSSSS      ....


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
```

```
0001    C
0002    C Version:      'V04-000'
0003    C
0004    C*****************************************************************************
0005    C*                                                                           *
0006    C*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
0007    C*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
0008    C*  ALL RIGHTS RESERVED.                                                     *
0009    C*                                                                           *
0010    C*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED    *
0011    C*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
0012    C*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER    *
0013    C*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY    *
0014    C*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY    *
0015    C*  TRANSFERRED.                                                             *
0016    C*                                                                           *
0017    C*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE    *
0018    C*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT    *
0019    C*  CORPORATION.                                                             *
0020    C*                                                                           *
0021    C*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS    *
0022    C*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
0023    C*                                                                           *
0024    C*                                                                           *
0025    C*****************************************************************************
0026    C
0027
0028    c       Author  Brian Porter            Creation Date   15-JUN-1981
0029
0030    c++
0031    c       Functional description:
0032    c
0033    c       This routine is called to identify the device for 'DEVICE ERROR',
0034    c       'DEVICE ATTENTION' and 'DEVICE TIMEOUT' entries.
0035    c
0036    c       Modified by:
0037    c
0038    C       V03-003 SAR0252         Sharon A. Reynolds      23-Apr-1984
0039    C               Added a field length for the output of the device name.
0040    C
0041    C       V03-002 SAR0068         Sharon A. Reynolds,     20-Jun-1983
0042    C               Changed the carriage control in the 'format' statements
0043    C               for use with ERF.
0044    C
0045    c       v03-001 BP0003          Brian Porter,           03-MAR-1982
0046    c               Added more arguments to dhead2 call.  Added dhead3
0047    c               routine
0048    c--
0049    c--
0050
0051
0052
0053
0054            subroutine dhead1 (lun,device_type)
0055
0056
0057
```

```
0058
0059            include 'src$:msghdr.for /nolist'
0118
0119            include 'src$:deverr.for /nolist'
0220
0221
0222
0223
0224            byte            lun
0225            character*(*)   device_type
0226
0227
0228
0229
0230            call header (lun)
0231
0232            if (emb$w_hd_entry .eq. 1) then
0233
0234            call logger (lun,'DEVICE ERROR')
0235
0236            else if (emb$w_hd_entry .eq. 96) then
0237
0238            call logger (lun,'DEVICE TIMEOUT')
0239
0240            else if (emb$w_hd_entry .eq. 98) then
0241
0242            call logger (lun,'DEVICE ATTENTION')
0243            endif
0244
0245            call dhead2 (lun,device_type,emb$b_dv_namlng,emb$t_dv_name,
0246            1 emb$w_dv_unit)
0247
0248            return
0249
0250            end
```

PROGRAM SECTIONS

	Name	Bytes	Attributes
0	$CODE	121	PIC CON REL LCL SHR EXE RD NOWRT LONG
1	$PDATA	45	PIC CON REL LCL SHR NOEXE RD NOWRT LONG
2	$LOCAL	108	PIC CON REL LCL NOSHR NOEXE RD WRT LONG
3	EMB	512	PIC OVR REL GBL SHR NOEXE RD WRT LONG
	Total Space Allocated	786	

ENTRY POINTS

Address	Type	Name
0-00000000		DHEAD1

VARIABLES

Address	Type	Name	Address	Type	Name
AP-00000008@	CHAR	DEVICE_TYPE	3-0000001C	L*1	EMB$B_DV_CLASS
3-00000010	L*1	EMB$B_DV_ERTCNT	3-00000011	L*1	EMB$B_DV_ERTMAX
3-0000003E	L*1	EMB$B_DV_NAMLNG	3-0000003A	L*1	EMB$B_DV_SLAVE
3-0000001D	L*1	EMB$B_DV_TYPE	3-00000036	I*4	EMB$L_DV_CHAR
3-00000012	I*4	EMB$L_DV_IOSB1	3-00000016	I*4	EMB$L_DV_IOSB2
3-00000026	I*4	EMB$L_DV_MEDIA	3-0000004E	I*4	EMB$L_DV_NUMREG
3-0000002E	I*4	EMB$L_DV_OPCNT	3-00000032	I*4	EMB$L_DV_OWNUIC
3-0000001E	I*4	EMB$L_DV_RQPID	3-00000000	I*4	EMB$L_HD_SID
3-0000003F	CHAR	EMB$T_DV_NAME	3-00000024	I*2	EMB$W_DV_BCNT
3-00000022	I*2	EMB$W_DV_BOFF	3-0000002C	I*2	EMB$W_DV_ERRCNT
3-0000003C	I*2	EMB$W_DV_FUNC	3-0000001A	I*2	EMB$W_DV_STS
3-0000002A	I*2	EMB$W_DV_UNIT	3-00000004	I*2	EMB$W_HD_ENTRY
3-0000000E	I*2	EMB$W_HD_ERRSEQ	AP-00000004@	L*1	LUN

ARRAYS

Address	Type	Name	Bytes	Dimensions
3-00000000	L*1	EMB	512	(0:511)
3-00000052	I*4	EMB$L_DV_REGSAV	420	(0:104)
3-00000006	I*4	EMB$Q_HD_TIME	8	(2)

FUNCTIONS AND SUBROUTINES REFERENCED

Type	Name	Type	Name	Type	Name
	DHEAD2		HEADER		LOGGER

```
      subroutine dhead2 (lun,device_type,emb$b_zz_namlng,emb$t_zz_name,
     1 emb$w_zz_unit)



      include 'src$:msghdr.for /nolist'



      byte            lun

      character*(*)   device_type

      byte            emb$b_zz_namlng

      character*(*)   emb$t_zz_name

      integer*2       emb$w_zz_unit

      integer*4       caller_mount_flag_and_label

      character*12    last_emb$t_vm_label

      integer*4       last_emb$t_vm_label_length

      logical*1       last_emb$t_vm_label_valid

      logical*1       str$trim

      integer*4       compress4



      last_emb$t_vm_label_valid = .false.

      call get_current_label (3,emb$l_hd_sid,emb$b_zz_namlng,emb$t_zz_name,
     1 emb$w_zz_unit,%ref(last_emb$t_vm_label),*5)

      last_emb$t_vm_label_valid = .true.

5     continue

      if (.not. str$trim (last_emb$t_vm_label,last_emb$t_vm_label,
     1 last_emb$t_vm_label_length)) then

      last_emb$t_vm_label_length = 12
      endif

      if (
     1 last_emb$t_vm_label_valid
```

```
0116            1 .and.
0117            1 last_emb$t_vm_label_length .gt. 1
0118            1 ) then
0119
0120     10     continue
0121
0122            call linchk (lun,2)
0123
0124            write(lun,15) device_type,emb$t_zz_name(1:emb$b_zz_namlng),
0125            1 emb$w_zz_unit,', CURRENT LABEL ""',
0126            1 last_emb$t_vm_label(1:last_emb$t_vm_label_length),'""'
0127     15     format(/' ',a,' SUB-SYSTEM, UNIT ',a<emb$b_zz_namlng>,
0128            1 i<compress4 (lib$extzv(0,16,emb$w_zz_unit))>,':',3(:a))
0129
0130            return
0131            endif
0132
0133     20     continue
0134
0135            call linchk (lun,2)
0136
0137            write(lun,15) device_type,emb$t_zz_name(1:emb$b_zz_namlng),
0138            1 emb$w_zz_unit
0139
0140            return
0141
0142
0143
0144            entry dhead3 (lun,device_type,emb$b_zz_namlng,emb$t_zz_name,
0145            1 emb$w_zz_unit,caller_mount_flag_and_label)
0146
0147
0148
0149            last_emb$t_vm_label_valid = .false.
0150
0151            if (caller_mount_flag_and_label .eq. -1) goto 20
0152
0153            call movc3 (%val(12),%val(caller_mount_flag_and_label),
0154            1 %ref(last_emb$t_vm_label))
0155
0156            if (.not. str$trim (last_emb$t_vm_label,last_emb$t_vm_label,
0157            1 last_emb$t_vm_label_length)) then
0158
0159            last_emb$t_vm_label_length = 12
0160            endif
0161
0162            last_emb$t_vm_label_valid = .true.
0163
0164            goto 10
0165
0166            end
```

PROGRAM SECTIONS

	Name	Bytes	Attributes
0	$CODE	449	PIC CON REL LCL SHR EXE RD NOWRT LONG
1	$PDATA	83	PIC CON REL LCL SHR NOEXE RD NOWRT LONG
2	$LOCAL	180	PIC CON REL LCL NOSHR NOEXE RD WRT LONG
3	EMB	512	PIC OVR REL GBL SHR NOEXE RD WRT LONG
	Total Space Allocated	1224	

ENTRY POINTS

Address	Type	Name	Address	Type	Name
0-00000000		DHEAD2	0-0000012B		DHEAD3

VARIABLES

Address	Type	Name	Address	Type	Name
AP-00000018@	I*4	CALLER_MOUNT_FLAG_AND_LABEL	AP-00000008@	CHAR	DEVICE_TYPE
2-00000014@	L*1	EMB$B_ZZ_NAMLNG	3-00000000	I*4	EMB$L_HD_SID
AP-00000010@	CHAR	EMB$T_ZZ_NAME	3-00000004	I*2	EMB$W_HD_ENTRY
3-0000000E	I*2	EMB$W_HD_ERRSEQ	2-00000018@	I*2	EMB$W_ZZ_UNIT
2-00000000	CHAR	LAST_EMB$T_VM_LABEL	2-00000010	I*4	LAST_EMB$T_VM_LABEL_LENGTH
2-0000000C	L*1	LAST_EMB$T_VM_LABEL_VALID	AP-00000004@	L*1	LUN

ARRAYS

Address	Type	Name	Bytes	Dimensions
3-00000000	L*1	EMB	512	(0:511)
3-00000006	I*4	EMB$Q_HD_TIME	8	(2)

LABELS

Address	Label	Address	Label	Address	Label	Address	Label
0-0000004A	5	0-00000062	10	1-00000022	15'	0-000000DA	20

FUNCTIONS AND SUBROUTINES REFERENCED

Type	Name	Type	Name	Type	Name
I*4	COMPRESS4		GET_CURRENT_LABEL	I*4	LIB$EXTZV
	LINCHK		MOVC3	L*1	STR$TRIM

DQD

COMMAND QUALIFIERS

```
FORTRAN /LIS=LIS$:DHEADS/OBJ=OBJ$:DHEADS MSRC$:DHEADS

/CHECK=(NOBOUNDS,OVERFLOW,NOUNDERFLOW)
/DEBUG=(NOSYMBOLS,TRACEBACK)
/STANDARD=(NOSYNTAX,NOSOURCE_FORM)
/SHOW=(NOPREPROCESSOR,NOINCLUDE,MAP)
/F77  /NOG_FLOATING  /I4  /OPTIMIZE  /WARNINGS  /NOD_LINES  /NOCROSS_REFERENCE  /NOMACHINE_CODE  /CONTINUATIONS=19
```

COMPILATION STATISTICS

```
Run Time:           2.96 seconds
Elapsed Time:       9.20 seconds
Page Faults:        132
Dynamic Memory:     175 pages
```

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

CLASSIFY
LIS

DR750
LIS

CSTRING
LIS

DHEADS
LIS

DR780
LIS

DR11W
LIS

COMPRESS
LIS

DTAILS
LIS

DECODECC
LIS

DUMPREG
LIS

OUTUDRIVR
LIS

CALCMAP
LIS

DUP3271
LIS

CRYPTK
LIS

DQDISKS
LIS

DUP11
LIS