

```

EEEEEEEEEEEEEEEEEE RRRRRRRRRRRR FFFFFFFF
EEEEEEEEEEEEEEEEEE RRRRRRRRRRRR FFFFFFFF
EEEEEEEEEEEEEEEEEE RRRRRRRRRRRR FFFFFFFF
EEE RRR RRR FFF
EEE RRR RRR FFF
EEE RRR RRR FFF
EEE RRR RRR FFF
EEE RRR RRR FFF
EEE RRR RRR FFF
EEEEEEEEEEEEEEEEEE RRRRRRRRRRRR FFFFFFFF
EEEEEEEEEEEEEEEEEE RRRRRRRRRRRR FFFFFFFF
EEEEEEEEEEEEEEEEEE RRRRRRRRRRRR FFFFFFFF
EEE RRR RRR FFF
EEE RRR RRR FFF
EEE RRR RRR FFF
EEE RRR RRR FFF
EEE RRR RRR FFF
EEE RRR RRR FFF
EEEEEEEEEEEEEEEEEE RRRRRRRRRRRR FFFF
EEEEEEEEEEEEEEEEEE RRRRRRRRRRRR FFFF
EEEEEEEEEEEEEEEEEE RRRRRRRRRRRR FFFF

```

[illegible]

```
UU      UU  NN  NN  KK  KK  NN  NN
UU      UU  NN  NN  KK  KK  NN  NN
UU      UU  NN  NN  KK  KK  NN  NN
UU      UU  NNNN NN  KK  KK  NNNN NN
UU      UU  NNNN NN  KK  KK  NNNN NN
UU      UU  NN  NN  NNNN KK  NN  NN
UU      UU  NN  NN  NNNN KK  NN  NN
UU      UU  NN  NN  NNNN KK  NN  NN
UU      UU  NN  NN  KK  KK  NN  NN
UUUUUUUU NN  NN  KK  KK  NN  NN
UUUUUUUU NN  NN  KK  KK  NN  NN

-----
-----

DDDDDDDD DD  IIIIII SSSSSSSS PPPPPPPP
DDDDDDDD DD  IIIIII SSSSSSSS PPPPPPPP
DD      DD  II  SS      PP      PP
DD      DD  II  SS      PP      PP
DD      DD  II  SS      PP      PP
DD      DD  II  SSSSSS PPPPPPPP
DD      DD  II  SSSSSS PPPPPPPP
DD      DD  II  SS      PP
DD      DD  II  SS      PP
DD      DD  II  SS      PP
DD      DD  II  SS      PP
DDDDDDDD IIIIII SSSSSSSS PP
DDDDDDDD IIIIII SSSSSSSS PP

.....
.....
.....
.....

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
LLLLLLLLL IIIIII SSSSSSSS
LLLLLLLLL IIIIII SSSSSSSS
```



```
1 0001 0 Module UNKN_DISP
2 0002 0 (%TITLE 'Unknown Dispatcher'
3 0003 0 IDENT = 'V04-000') =
4 0004 0
5 0005 1 Begin
6 0006 1
7 0007 1
8 0008 1 *****
9 0009 1 *
10 0010 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
11 0011 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
12 0012 1 * ALL RIGHTS RESERVED.
13 0013 1 *
14 0014 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
15 0015 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
16 0016 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
17 0017 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
18 0018 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
19 0019 1 * TRANSFERRED.
20 0020 1 *
21 0021 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
22 0022 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
23 0023 1 * CORPORATION.
24 0024 1 *
25 0025 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
26 0026 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
27 0027 1 *
28 0028 1 *
29 0029 1 *****
30 0030 1
31 0031 1 ++
32 0032 1 FACILITY: ERF, Error Log Report Generator
33 0033 1
34 0034 1 ABSTRACT:
35 0035 1
36 0036 1 This module contains the dispatching routines for outputing
37 0037 1 'unknown' entries.
38 0038 1
39 0039 1 ENVIRONMENT:
40 0040 1
41 0041 1 VAX/VMS operating system, user mode.
42 0042 1
43 0043 1 AUTHOR: Sharon Reynolds, CREATION DATE: April 1983
44 0044 1
45 0045 1 Modified By:
46 0046 1
47 0047 1 V03-003 EAD0190 Elliott A. Drayton 6-Jul-1984
48 0048 1 Obtain the value of LSTLUN from SYECOM.
49 0049 1
50 0050 1 V03-002 EAD0143 Elliott A. Drayton 12-Apr-1984
51 0051 1 Remove reference to EMBETDEF.REQ.
52 0052 1
53 0053 1 V03-001 SAR0018 Sharon A. Reynolds, 22-Apr-1983
54 0054 1 Changed references to valid class, valid cpu, valid entry,
55 0055 1 and valid_type so they would be accessed via SYECOM.
56 0056 1
57 0057 1 --
```

UNKN_DISP
V04-000

Unknown Dispatcher

L 16
15-Sep-1984 23:55:50
14-Sep-1984 12:28:19

VAX-11 BLISS-32 V4.0-742
[ERF.SRC]UNKN_DISP.B32;1

Page 2
(1)

```

: 58      0058 1
: 59      0059 1
: 60      0060 1  ! Require files
: 61      0061 1
: 62      0062 1  ! Require 'SRC$:RECSELDEF.REQ' ;      ! EMB field definitions
: 63      0193 1  ! Require 'SRC$:ERFDEF.REQ' ;      ! ERF common definitions
: 64      0479 1
: 65      0480 1  Forward routine
: 66      0481 1      Unknown_dispatcher: NOVALUE ;
: 67      0482 1
: 68      0483 1
: 69      0484 1  External routine
: 70      0485 1      Dumpreg,
: 71      0486 1      Header,
: 72      0487 1      Logger,
: 73      0488 1      Unknown1,
: 74      0489 1      Unknown2,
: 75      0490 1      Unknown3 ;
: 76      0491 1
: 77      0492 1
: 78      0493 1  External
: 79      0494 1      Input_rab:      $BBLOCK [],
: 80      0495 1      Syecom:      $BBLOCK PSECT (SYECOM);
: 81      0496 1
: 82      0497 1  Own
: 83      0498 1      Temp_buf:      PSECT(EMB),
: 84      0499 1      Emb_buf:      Initial (temp_buf)
: 85      0500 1      REF $BBLOCK,
: 86      0501 1      Unknown_entry: LONG,
: 87      0502 1      Unknown_device: LONG,
: 88      0503 1      Unknown_cpu: LONG;
: 89      0504 1
: 90      0505 1
```



```

: 92      0506 1 Global Routine UNKNOWN_DISPATCHER: NOVALUE =
: 93      0507 2 Begin
: 94      0508 2
: 95      0509 2 ++
: 96      0510 2
: 97      0511 2 Functional Description:
: 98      0512 2
: 99      0513 2 This routine dispatches to any module necessary for the output
100      0514 2 of 'unknown' entries.
101      0515 2
102      0516 2 Calling Sequence:
103      0517 2
104      0518 2 Unknown_dispatcher ()
105      0519 2
106      0520 2 Input Parameters:
107      0521 2
108      0522 2
109      0523 2 Output Parameters:
110      0524 2
111      0525 2 None
112      0526 2
113      0527 2 --
114      0528 2
115      0529 2
116      0530 2 Record_size = Size (bytes) of the entry being processed
117      0531 2
118      0532 2 Bind record_size = input_rab[rab$w_rsz] ;
119      0533 2
120      0534 2 Unknown_entry = $descriptor('UNKNOWN ENTRY') ;
121      0535 2 Unknown_device = $descriptor('UNKNOWN DEVICE' ENTRY') ;
122      0536 2 Unknown_cpu = $descriptor('UNKNOWN CPU' ENTRY') ;
123      0537 2
124      0538 2
125      0539 2 Output the first two lines of the entry header.
126      0540 2
127      0541 2 HEADER (syecom[sye$l_lstlun]) ;
128      0542 2
129      0543 2
130      0544 2 Determine what message to output in the entry header.
131      0545 2
132      0546 3 If (.syecom[sye$b_valid_cpu] AND (NOT .syecom[sye$b_valid_entry]))
133      0547 2 Then
134      0548 2
135      0549 2 Indicate that this is an unknown entry type.
136      0550 2
137      0551 2 LOGGER (syecom[sye$l_lstlun], .unknown_entry)
138      0552 2 Else
139      0553 3 Begin
140      0554 4 If (.syecom[sye$b_valid_cpu] AND
141      0555 5 (.emb_buf[emb$w_hd_entry] EQL EMB$C_DE OR
142      0556 5 .emb_buf[emb$w_hd_entry] EQL EMB$C_DT OR
143      0557 4 .emb_buf[emb$w_hd_entry] EQL EMB$C_DA) OR
144      0558 4 (NOT .syecom[sye$b_valid_class]) OR
145      0559 4 (NOT .syecom[sye$b_valid_type]))
146      0560 3 Then
147      0561 3
148      0562 3 Indicate that this is an unknown device.

```

```

! Device error 1
! Device timeout 96 %x60
! Device attention 98 %x62

```



```

149 0563 3      !
150 0564 3      ! LOGGER (syecom[sye$l_lstlun], .unknown_device)
151 0565 3      Else
152 0566 4      Begin
153 0567 5      If (NOT .syecom[sye$b_valid_cpu])
154 0568 4      Then
155 0569 4      |
156 0570 4      |   Indicate that this is an unknown cpu type.
157 0571 4      |
158 0572 4      |   LOGGER (syecom[sye$l_lstlun], .unknown_cpu) ;
159 0573 3      |   End ;
160 0574 3      End ;
161 0575 2
162 0576 2
163 0577 2      |
164 0578 2      |   Output the contents of the header packet.
165 0579 2      |
166 0580 2      |   UNKNOWN1 (syecom[sye$l_lstlun]) ;
167 0581 2
168 0582 2      |
169 0583 2      |   Determine what else in the entry can be translated and call the
170 0584 2      |   appropriate routine to do so.
171 0585 2      |
172 0586 2      |   If (.emb_buf[emb$w_hd_entry] EQL EMB$C_DE) OR      ! Device error 1
173 0587 2      |   (.emb_buf[emb$w_hd_entry] EQL EMB$C_DT)          ! Device timeout 96 %x60
174 0588 2      |   Then
175 0589 2      |   |
176 0590 2      |   |   UNKNOWN2 (syecom[sye$l_lstlun])
177 0591 2      |   Else
178 0592 2      |   |
179 0593 2      |   |   Begin
180 0594 4      |   |   |   If (.emb_buf[emb$w_hd_entry] EQL EMB$C_DA)      ! Device attention 98 %x62
181 0595 4      |   |   |   Then
182 0596 4      |   |   |   |
183 0597 4      |   |   |   |   UNKNOWN3 (syecom[sye$l_lstlun])
184 0598 4      |   |   |   |   Else
185 0599 4      |   |   |   |   |
186 0600 4      |   |   |   |   |   Nothing else can be translated, output the remainder of the
187 0601 4      |   |   |   |   |   entry in a longword, hexadecimal format.
188 0602 4      |   |   |   |   |   |
189 0603 4      |   |   |   |   |   |   DUMPREG (syecom[sye$l_lstlun], %REF((((record_size + 3)/4) - 4)),
190 0604 4      |   |   |   |   |   |   emb_buf[16.0,0,0]) ;
191 0605 4      |   |   |   |   |   |   End ;
192 0606 4      |   |   |   |   |   |   End ;
193 0607 2      |   |   |   |   |   |   End ;
194 0608 2      |   |   |   |   |   |   End ;
195 0609 2      |   |   |   |   |   |   End ;
196 0610 1      |   |   |   |   |   |   End ; ! Routine
```

```
.TITLE UNKN_DISP Unknown Dispatcher
.IDENT \V04-000\
```

```
.PSECT $PLIT,NOWRT,NOEXE, PIC,2
```

```
22 59 52 54 4E 45 20 4E 57 4F 4E 4B 4E 55 22 00000 P.AAB: .ASCII \ 'UNKNOWN ENTRY' \
0000F .BLKB 1 ;
```


[illegible]

	04	F2	08	13	0005A	BEQL	3\$		
	05	F5	A3	E9	0005C	2\$: BLBC	SYECOM+25, 3\$		0558
		04	A3	E8	00060	BLBS	SYECOM+28, 4\$		0559
			A4	DD	00064	3\$: PUSHL	UNKNOWN_DEVICE		0564
			06	11	00067	BRB	5\$		
	0C		52	E8	00069	4\$: BLBS	R2, 6\$		0567
		08	A4	DD	0006C	PUSHL	UNKNOWN_CPU		0572
			53	DD	0006F	5\$: PUSHL	R3		
00000000G	00		02	FB	00071	CALLS	#2, LOGGER		
			53	DD	00078	6\$: PUSHL	R3		0579
00000000G	00		01	FB	0007A	CALLS	#1, UNKNOWN1		
	52	FC	A4	DD	00081	MOVL	EMB_BUF, R2		0585
	01	04	A2	B1	00085	CMPW	4(R2), #1		
			08	13	00089	BEQL	7\$		
0060	8F	04	A2	B1	0008B	CMPW	4(R2), #96		0586
			0A	12	00091	BNEQ	8\$		
			53	DD	00093	7\$: PUSHL	R3		0591
00000000G	00		01	FB	00095	CALLS	#1, UNKNOWN2		
			04	0009C	RET				
0062	8F	04	A2	B1	0009D	8\$: CMPW	4(R2), #98		0594
			0A	12	000A3	BNEQ	9\$		
			53	DD	000A5	PUSHL	R3		0599
00000000G	00		01	FB	000A7	CALLS	#1, UNKNOWN3		
			04	000AE	RET				
		10	A2	9F	000AF	9\$: PUSHAB	16(R2)		0606
50 00000000G	00		03	C1	000B2	ADDL3	#3, RECORD_SIZE, R0		0605
	50		04	C6	000BA	DIVL2	#4, R0		
	04	AE	FC	A0	9E	000BD	MOVAB	-4(R0), 4(SP)	
		04	AE	9F	000C2	PUSHAB	4(SP)		
			53	DD	000C5	PUSHL	R3		
00000000G	00		03	FB	000C7	CALLS	#3, DUMPREG		0606
			04	000CE	RET				0610

; Routine Size: 207 bytes, Routine Base: \$CODE + 0000

:	197	0611	1	
:	198	0612	1	End ! Module
:	199	0613	0	ELUDOM

PSECT SUMMARY

Name	Bytes	Attributes
EMB	4	NOVEC, WRT, RD, NOEXE, SHR, GBL, REL, OVR, PIC, ALIGN(2)
\$OWNS	16	NOVEC, WRT, RD, NOEXE, NOSHR, LCL, REL, CON, PIC, ALIGN(2)
\$PLIT	84	NOVEC, NOWRT, RD, NOEXE, NOSHR, LCL, REL, CON, PIC, ALIGN(2)
\$CODE	207	NOVEC, NOWRT, RD, EXE, NOSHR, LCL, REL, CON, PIC, ALIGN(2)

UNKN_DISP
V04-000

Unknown Dispatcher

E 1
15-Sep-1984 23:55:50
14-Sep-1984 12:28:19

VAX-11 Bliss-32 V4.0-742
[ERF.SRC]UNKN_DISP.B32;1

Page 7
(2)

Library Statistics

File	----- Total	Symbols Loaded	----- Percent	Pages Mapped	Processing Time
_\$255\$DUA28:[SYSLIB]LIB.L32;1	18619	26	0	1000	00:02.0

COMMAND QUALIFIERS

BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS\$:UNKN_DISP/OBJ=OBJ\$:UNKN_DISP MSRC\$:UNKN_DISP/UPDATE=(ENH\$:UNKN_DISP)

: Size: 207 code + 104 data bytes
: Run Time: 00:09.0
: Elapsed Time: 00:20.2
: Lines/CPU Min: 4109
: Lexemes/CPU-Min: 24382
: Memory Used: 108 pages
: Compilation Complete

0154 AH-BT13A-SE DIGITAL EQUIPMENT CORPORATION
VAX/VMS V4.0 CONFIDENTIAL AND PROPRIETARY

0154 AH-BT13A-SE DIGITAL EQUIPMENT CORPORATION
VAX/VMS V4.0 CONFIDENTIAL AND PROPRIETARY

0155 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY.