

[illegible][illegible]

MEM

047
047
047
047
047
047
047
047
047
048
048
048
048
048
048
048
048
049
049
049
049
049
049
049
049
049
049
050
050
050
050
050
050
050
050
050
050
051
051
051
051
051
051
051
051
051
051
051
052
052
052
052
052
052
052

B 2
15-Sep-1984 23:51:45
14-Sep-1984 12:27:43

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

Page 1
(1)

```
0001 0 MODULE MCHK_DISP
0002 0 (%TITLE 'Machine check dispatcher'
0003 0 IDENT = 'V04-000') =
0004 0
0005 1 BEGIN
0006 1
0007 1
0008 1 *****
0009 1 *
0010 1 *  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0011 1 *  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0012 1 *  ALL RIGHTS RESERVED.
0013 1 *
0014 1 *  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0015 1 *  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0016 1 *  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0017 1 *  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0018 1 *  OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0019 1 *  TRANSFERRED.
0020 1 *
0021 1 *  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0022 1 *  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0023 1 *  CORPORATION.
0024 1 *
0025 1 *  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0026 1 *  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0027 1 *
0028 1 *
0029 1 *****
0030 1
0031 1 ++
0032 1 FACILITY: ERLRPT, Error Log Report Generator
0033 1
0034 1 ABSTRACT:
0035 1
0036 1     This module contains the machine check dispatching routine.
0037 1
0038 1 ENVIRONMENT:
0039 1
0040 1     VAX/VMS operating system, user mode.
0041 1
0042 1 AUTHOR: Sharon A. Reynolds, CREATION DATE: 10-Apr-1984
0043 1
0044 1 Modified by:
0045 1
0046 1     V04-003 SAR0290 Sharon A. Reynolds 17-Jul-1984
0047 1     Changed the way that option is referenced due to change
0048 1     in the way it is passed.
0049 1
0050 1     V04-002 SAR0256 Sharon A. Reynolds 25-apr-1984
0051 1     Put in check to determine whether to output text.
0052 1
0053 1     V04-001 EAD0139 Elliott A. Drayton 12-Apr-1984
0054 1     Removed reference to LIB.REQ.
0055 1
0056 1 --
0057 1
```



```

58      0058      1  REQUIRE 'SRC$:ERFDEF.REQ' ;
59      0344      1
60      0345      1  MACRO
61      0346      1      Cpu_fld = 0,24,8,0 % ;
62      0347      1
63      0348      1  FORWARD ROUTINE
64      0349      1      Mchk_dispatcher:      Novalue ;
65      0350      1
66      0351      1  EXTERNAL ROUTINE
67      0352      1      Frctof,
68      0353      1      Header,
69      0354      1      Logger,
70      0355      1      Mcheck_7ZZ,
71      0356      1      Mcheck_750,
72      0357      1      Mcheck_780,
73      0358      1      Mcheck_UV1 ;
74      0359      1
75      0360      1  EXTERNAL
76      0361      1      Emb:      $BBLOCK PSECT (emb) ;
77      0362      1

```

MEMO

058
058
058
058
059
059
059
059
059
059
059
059
060
060
060
060
060
060
060
060
061
061
061
061
061
061
061
061
061
061
062
062
062
062
062
062
062
062
062
063
063
063
063
063
063
063
063
064
064


```

79 0363 1 GLOBAL Routine MCHK_DISPATCHER (lun,option) : NOVALUE =
80 0364 2 Begin
81 0365 2
82 0366 2 ++
83 0367 2
84 0368 2 Functional Description:
85 0369 2
86 0370 2 This routine outputs the machine check header, then via the
87 0371 2 cpu-type determines which cpu dependent routine to invoke and
88 0372 2 invokes the routine.
89 0373 2
90 0374 2 Calling Sequence:
91 0375 2
92 0376 2 MCHK_DISPATCHER (lun,option)
93 0377 2
94 0378 2 Input Parameters:
95 0379 2
96 0380 2 Lun = logical unit number of output device.
97 0381 2 Option = address of descriptor that contains the option.
98 0382 2
99 0383 2 Output Parameters:
100 0384 2
101 0385 2 None.
102 0386 2
103 0387 2 --
104 0388 2
105 0389 2 LOCAL
106 0390 2 Cpu_type: REF $BBLOCK ;
107 0391 2
108 0392 2 Map
109 0393 2 Option: REF $BBLOCK ;
110 0394 2
111 0395 2 Bind start_of_cpu_specific_stuff = emb[emb$b_mc_sumcod] ;
112 0396 2
113 0397 2
114 0398 2 Determine whether to output any text.
115 0399 2
116 0400 2 If ..option[dsc$a_pointer] NEQ 'S'
117 0401 2 Then
118 0402 2 Return ;
119 0403 2
120 0404 2
121 0405 2 Force a top of form to occur, and output the entry header.
122 0406 2
123 0407 2 FRCTOF (.lun) ;
124 0408 2 HEADER (.lun) ;
125 0409 2 LOGGER (.lun,%ASCID 'MACHINE CHECK') ;
126 0410 2
127 0411 2
128 0412 2 Get the cpu type field from the system id register.
129 0413 2
130 0414 2 Cpu_type = emb[emb$l_hd_sid] ;
131 0415 2 Cpu_type = .cpu_type[cpu_fld] ;
132 0416 2
133 0417 2
134 0418 2 Determine if the cpu type is an 11/780 without the cpu field
135 0419 2 in the system id set.
```



```

136      0420      2  !
137      0421      2  if .cpu_type EQLU 255
138      0422      2  Then
139      0423      2  | Yes, call the 11/780 machine check module to output
140      0424      2  | the rest of the report.
141      0425      2  |
142      0426      2  | MCHECK_780 (.lun,start_of_cpu_specific_stuff) ;
143      0427      2  |
144      0428      2  |
145      0429      2  | Determine if the cpu type is any other valid type. If a valid/supported
146      0430      2  | cpu type is found then call the cpu dependent machine check
147      0431      2  | module.
148      0432      2  |
149      0433      2  Case .cpu_type from 1 to 7 of
150      0434      2  Set
151      0435      2  [1]: MCHECK_780 (.lun,start_of_cpu_specific_stuff) ;
152      0436      2  [2]: MCHECK_750 (.lun,start_of_cpu_specific_stuff) ;
153      0437      2  [3]: MCHECK_7ZZ (.lun,start_of_cpu_specific_stuff) ;
154      0438      2  [7]: MCHECK_UV1 (.lun,start_of_cpu_specific_stuff) ;
155      0439      2  [INRANGE]: Return ;
156      0440      2  Tes ;
157      0441      2  [INRANGE]: Return ;
158      0442      2  Tes ;
159      0443      2  [INRANGE]: Return ;
160      0444      2  Tes ;
161      0445      2  [INRANGE]: Return ;
162      0446      2  End ;      ! Routine

```

```
.TITLE MCHK_DISP Machine check dispatcher
.IDENT \V04=000\
```

```
.PSECT SPLIT,NOWRT,NOEXE, PIC,2
```

```
00 00 4B 43 45 48 43 20 45 4E 49 48 43 41 4D 00000 P.AAB: .ASCII \MACHINE CHECK\<0><0><0>
```

```

00 0000F
010E000D 00010 P.AAA: .LONG 17694733
00000000' 00014 .ADDRESS P.AAB

```

```
.EXTRN  FRCTOF, HEADER, LOGGR
.EXTRN  MCHECK_722, MCHECK_750
.EXTRN  MCHECK_780, MCHECK_UV1
.EXTRN  EMB
```

```
.PSECT $CODE,NOWRT, PIC,2
```

					003C	00000
	55	00000000G	00		9E	00002
	54	00000000G	00		9E	00009
	50	08	AC	D0		00010
00000053	8F	04	B0	D1		00014
			70	12		0001C
	53	04	AC	D0		0001E
			53	DD		00022
00000000G	00		01	FB		00024
			53	DD		0002B
00000000G	00		01	FB		0002D

```

.ENTRY      MCHK_DISPATCHER, Save R2,R3,R4,R5
MOVAB      MCHECK 780, R5
MOVAB      START OF CPU_SPECIFIC_STUFF, R4
MOVL      OPTION, R0
CML       @4(R0), #83
BNEQ      7$
MOVL      LUN, R3
PUSHL     R3
CALLS     #1, FRCTOF
PUSHL     R3
CALLS     #1, HEADER

```

0363
0400
0407
0408

MCHK_DISP
V04-000

Machine check dispatcher

F 2
15-Sep-1984 23:51:45
14-Sep-1984 12:27:43

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

Page 5
(2)

		00000000'	00	9F	00034	PUSHAB	P,AAA	0409
			53	DD	0003A	PUSHL	R3	
	00000000G	00	02	FB	0003C	CALLS	#2, LOGGER	
		52	A4	9E	00043	MOVAB	EMB, CPU_TYPE	0414
		52	A2	9A	00047	MOVZBL	3(CPU_TYPE), CPU_TYPE	0415
	000000FF	8F	52	D1	0004B	CMPL	CPU_TYPE, #255	0421
			05	12	00052	BNEQ	1\$	
		65	18	BB	00054	PUSHR	#^M<R3,R4>	0426
		01	02	FB	00056	CALLS	#2, MCHECK_780	
0031	001E	0014	52	CF	00059	CASEL	CPU_TYPE, #1, #6	0433
	0028	0031	000E		0005D	.WORD	3\$-2\$,-	
			0031		00065		4\$-2\$,-	
							5\$-2\$,-	
							7\$-2\$,-	
							7\$-2\$,-	
							7\$-2\$,-	
							6\$-2\$	
			18	BB	0006B	PUSHR	#^M<R3,R4>	0435
		65	02	FB	0006D	CALLS	#2, MCHECK_780	
				04	00070	RET		
			18	BB	00071	PUSHR	#^M<R3,R4>	0437
	00000000G	00	02	FB	00073	CALLS	#2, MCHECK_750	
				04	0007A	RET		
			18	BB	0007B	PUSHR	#^M<R3,R4>	0439
	00000000G	00	02	FB	0007D	CALLS	#2, MCHECK_722	
				04	00084	RET		
			18	BB	00085	PUSHR	#^M<R3,R4>	0441
	00000000G	00	02	FB	00087	CALLS	#2, MCHECK_UV1	
				04	0008E	RET		0446

; Routine Size: 143 bytes, Routine Base: \$CODE + 0000

:	163	0447	1		
:	164	0448	1	End	! Module
:	165	0449	0	ELUDOM	

PSECT SUMMARY

Name	Bytes	Attributes
\$PLIT	24	NOVEC,NOWRT, RD ,NOEXE,NOSHR, LCL, REL, CON, PIC,ALIGN(2)
\$CODE	143	NOVEC,NOWRT, RD , EXE,NOSHR, LCL, REL, CON, PIC,ALIGN(2)

Library Statistics

File	----- Total	Symbols Loaded	----- Percent	Pages Mapped	Processing Time
------	----------------	-------------------	------------------	-----------------	--------------------

MCHK_DISP
V04-000

Machine check dispatcher

6 2
15-Sep-1984 23:51:45
14-Sep-1984 12:27:43

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

Page 6
(2)

: _\$255\$DUA28:[SYSLIB]LIB.L32;1

18619

23

0

1000

00:01.9

: COMMAND QUALIFIERS

: BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LISS:MCHK_DISP/OBJ=OBJ\$:MCHK_DISP MSRC\$:MCHK_DISP/UPDATE=(ENHS:MCHK_DISP)

: Size: 143 code + 24 data bytes
: Run Time: 00:07.1
: Elapsed Time: 00:14.5
: Lines/CPU Min: 3810
: Lexemes/CPU-Min: 21801
: Memory Used: 84 pages
: Compilation Complete

0151 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

Grid of 160 (10 rows by 16 columns) small panels, each containing technical data, diagrams, and labels. Key labels include:

- MESSAGE LIS
- ML11 LIS
- MSOP LIS
- MTAPE LIS
- MOUNT LIS
- MOUXX LIS
- MEMORYS LIS
- MCHK.DISP LIS