


```

DDDDDDDD      IIIIII      SSSSSSSS  PPPPPPPP      IIIIII      MM      MM      DDDDDDDD      MM      MM      PPPPPPPP
DDDDDDDD      IIIIII      SSSSSSSS  PPPPPPPP      IIIIII      MM      MM      DDDDDDDD      MM      MM      PPPPPPPP
DD      DD      II      SS      PP      PP      II      MMMM  MMMM  DD      DD      MMMM  MMMM  PP      PP
DD      DD      II      SS      PP      PP      II      MMMM  MMMM  DD      DD      MMMM  MMMM  PP      PP
DD      DD      II      SS      PP      PP      II      MM  MM  MM  DD      DD      MM  MM  MM  PP      PP
DD      DD      II      SSSSSS  PPPPPPPP      II      MM      MM  DD      DD      MM      MM  PPPPPPPP
DD      DD      II      SSSSSS  PPPPPPPP      II      MM      MM  DD      DD      MM      MM  PPPPPPPP
DD      DD      II      SS      PP      PP      II      MM      MM  DD      DD      MM      MM  PP
DD      DD      II      SS      PP      PP      II      MM      MM  DD      DD      MM      MM  PP
DD      DD      II      SS      PP      PP      II      MM      MM  DD      DD      MM      MM  PP
DD      DD      II      SS      PP      PP      II      MM      MM  DD      DD      MM      MM  PP
DDDDDDDD      IIIIII      SSSSSSSS  PP      PP      IIIIII  MM      MM  DDDDDDDD      MM      MM  PP
DDDDDDDD      IIIIII      SSSSSSSS  PP      PP      IIIIII  MM      MM  DDDDDDDD      MM      MM  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
LLLLLLLLLLLL  IIIIII      SSSSSSSS
LLLLLLLLLLLL  IIIIII      SSSSSSSS

```

```

1 0001 0 XTITLE ' Display image dump data'
2 0002 0 MODULE DISPIMDMP ( LANGUAGE (BLISS32),
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
33 0033 1 FACILITY: Image Dump
34 0034 1
35 0035 1 ABSTRACT:
36 0036 1 This module contains routines for analyzing and displaying information
37 0037 1 about image dumps.
38 0038 1
39 0039 1 ENVIRONMENT:
40 0040 1
41 0041 1 VAX/VMS Operating System
42 0042 1
43 0043 1 --
44 0044 1
45 0045 1
46 0046 1 AUTHOR: Wayne M. Cardoza, CREATION DATE: 10-Feb-1983
47 0047 1
48 0048 1 MODIFIED BY:
49 0049 1
50 0050 1 V03-005 WMC0005 Wayne Cardoza 04-Jun-1984
51 0051 1 Display patch data only when relevant.
52 0052 1
53 0053 1 V03-004 WMC0004 Wayne Cardoza 01-Sep-1983
54 0054 1 Add identifying data for the original image.
55 0055 1
56 0056 1 V03-003 WMC0003 Wayne Cardoza 06-Jul-1983
57 0057 1 Support /OUTPUT.

```



```
67 0066 1 !
68 0067 1 LIBRARY 'SYSS$LIBRARY:LIB';
69 0068 1 REQUIRE 'LIBS:$IMGDMPDEF.R32';
70 0194 1
71 0195 1
72 0196 1
73 0197 1 OWN
74 P 0198 1   OUTFAB      : $FAB (FNM = 'SYSS$OUTPUT',      ! FAB for SYSS$OUTPUT
75 0199 1       FAC = <GET,PUT>),
76 0200 1   OUTRAB      : $RAB (FAB = OUTFAB)
77 0201 1   PROMPT     : VECTOR[%CHARCOUNT ('Press RETURN to continue: ') + 2,BYTE]
78 0202 1       INITIAL(BYTE(13,10,'Press RETURN to continue: ')),
79 0203 1   ANSWER     : VECTOR[4,BYTE],                ! Buffer to hold responses
80 P 0204 1   INFAB      : $FAB (FNM = 'SYSS$INPUT',        ! FAB for SYSS$INPUT
81 0205 1       FAC = <GET,PUT>),
82 P 0206 1   INRAB      : $RAB (FAB = INFAB,
83 P 0207 1       PBF = PROMPT,
84 P 0208 1       PSZ = %CHARCOUNT ('Press RETURN to continue: ') + 2,
85 P 0209 1       UBF = ANSWER,
86 P 0210 1       USZ = 4,
87 0211 1       ROP = PMT),
88 0212 1   DSPBUF     : VECTOR[512,BYTE],                ! Buffer for text to be displayed
89 0213 1   DSPDESC    : VECTOR[2] INITIAL(512,DSPBUF),
90 0214 1   DSPLLEN;
91 0215 1
92 0216 1
93 0217 1
94 0218 1 EXTERNAL
95 0219 1   IMGHDR      : BLOCK[,BYTE],                ! Image header from the dump (first block)
96 0220 1   MISC        : BLOCK[,BYTE],                ! Buffer holding the misc data block
97 0221 1   MISC_VA     : BLOCK[,BYTE];                ! VA's of misc P1 space
98 0222 1
99 0223 1
100 0224 1
101 0225 1 EXTERNAL ROUTINE
102 0226 1   CLISGET VALUE,
103 0227 1   CLISPRESENT;
104 0228 1
105 0229 1
106 0230 1
107 0231 1 FORWARD ROUTINE
108 0232 1   DISPLAY_RELOC,
109 0233 1   DISPLAY_MISC,
110 0234 1   CHECK_INTER : NOVALUE;
111 0235 1
112 0236 1
113 0237 1
114 0238 1 MACRO
115 M 0239 1   QUIT ON ERROR =
116 M 0240 1     IF NOT .RETSTATUS
117 M 0241 1     THEN
118 0242 1       RETURN .RETSTATUS; %;
119 0243 1
120 0244 1 KEYWORDMACRO
121 M 0245 1   DISPLAY_FIELD (STRING, FIELD1=0, FIELD2=0, FIELD3=0, FIELD4=0 ) =
122 M 0246 1     BEGIN
123 M 0247 1     RETSTATUS =
```

```
.. 124      M 0248 1      $FAO ( %ASCID %STRING(%CHAR(13,10),STRING),
.. 125      M 0249 1          DSPLEN,
.. 126      M 0250 1          DSPDESC,
.. 127      M 0251 1          FIELD1,
.. 128      M 0252 1          FIELD2,
.. 129      M 0253 1          FIELD3,
.. 130      M 0254 1          FIELD4);
.. 131      M 0255 1      QUIT ON ERROR;
.. 132      M 0256 1      OUTRAB[RAB$L_RBF] = DSPBUF;
.. 133      M 0257 1      OUTRAB[RAB$W_RSZ] = .DSPLEN;
.. 134      M 0258 1      RETSTATUS =
.. 135      M 0259 1      $PUT (RAB = OUTRAB);
.. 136      M 0260 1      QUIT_ON_ERROR;
.. 137      M 0261 1      END;
.. 138      M 0262 1      %;
.. 139      M 0263 1
.. 140      M 0264 1      KEYWORDMACRO
.. 141      M 0265 1      ADD_FIELD (STRING, FIELD1=0, FIELD2=0, FIELD3=0) =
.. 142      M 0266 1      _BEGIN
.. 143      M 0267 1      RETSTATUS =
.. 144      M 0268 1      $FAO ( %ASCID %STRING(STRING),
.. 145      M 0269 1          DSPLEN,
.. 146      M 0270 1          DSPDESC,
.. 147      M 0271 1          FIELD1,
.. 148      M 0272 1          FIELD2,
.. 149      M 0273 1          FIELD3);
.. 150      M 0274 1      QUIT ON ERROR;
.. 151      M 0275 1      OUTRAB[RAB$L_RBF] = DSPBUF;
.. 152      M 0276 1      OUTRAB[RAB$W_RSZ] = .DSPLEN;
.. 153      M 0277 1      RETSTATUS =
.. 154      M 0278 1      $PUT (RAB = OUTRAB);
.. 155      M 0279 1      QUIT_ON_ERROR;
.. 156      M 0280 1      END
.. 157      M 0281 1      %;
```

```
159 0282 1 GLOBAL ROUTINE DISPLAY_DUMP =
160 0283 1
161 0284 1 +-+
162 0285 1
163 0286 1 Functional Description:
164 0287 1 This is the main routine for displaying a dump file.
165 0288 1
166 0289 1 Calling Sequence:
167 0290 1 standard
168 0291 1
169 0292 1 Input Parameters:
170 0293 1 none
171 0294 1
172 0295 1 Implicit Inputs:
173 0296 1 command line
174 0297 1
175 0298 1 Output Parameters:
176 0299 1 none
177 0300 1
178 0301 1 Implicit Outputs:
179 0302 1 none
180 0303 1
181 0304 1 Routines Called:
182 0305 1 none
183 0306 1
184 0307 1 Routine Value:
185 0308 1 none
186 0309 1
187 0310 1 Signals:
188 0311 1 none
189 0312 1
190 0313 1 Side Effects:
191 0314 1 none
192 0315 1
193 0316 1 --
194 0317 2 BEGIN
195 0318 2 LOCAL
196 0319 2 FULL, ! indicate if /FULL was requested
197 0320 2 OUTNAME : BLOCK [4,BYTE],
198 0321 2 RETSTATUS;
199 0322 2
200 0323 2 : Get output , lename
201 0324 2
202 0325 2 IF CLISPRESNT (%ASCID 'OUTPUT')
203 0326 2 THEN
204 0327 3 BEGIN
205 0328 3 OUTNAME[DSC$B_CLASS] = DSC$K_CLASS_D;
206 0329 3 OUTNAME[DSC$B_DTYPE] = DSC$K_DTYPE_T;
207 0330 3 OUTNAME[DSC$W_LENGTH] = 0;
208 0331 3 OUTNAME[DSC$A_POINTER] = 0;
209 0332 3 RETSTATUS =
210 0333 3 CLISGET_VALUE (%ASCID 'OUTPUT', OUTNAME);
211 0334 3 QUIT ON_ERROR;
212 0335 3 OUTFAB[FAB$L_FNA] = .OUTNAME[DSC$A_POINTER];
213 0336 3 OUTFAB[FAB$B_FNS] = .OUTNAME[DSC$W_LENGTH];
214 0337 2 END;
215 0338 2
```

```

216 0339 2 ! Set up for IO
217 0340 2 !
218 0341 2 RETSTATUS =
219 0342 2 $CREATE (FAB = OUTFAB);
220 0343 2 QUIT_ON_ERROR;
221 0344 2 RETSTATUS =
222 0345 2 $CONNECT(RAB = OUTRAB);
223 0346 2 QUIT_ON_ERROR;
224 0347 2 RETSTATUS =
225 0348 2 $OPEN (FAB = INFAB);
226 0349 2 QUIT_ON_ERROR;
227 0350 2 RETSTATUS =
228 0351 2 $CONNECT(RAB = INRAB);
229 0352 2 QUIT_ON_ERROR;
230 0353 2 !
231 0354 2 ! Display the data if requested
232 0355 2 !
233 0356 2 FULL = CLISPRESNT( %ASCID 'FULL');
234 0357 2 !
235 0358 2 IF CLISPRESNT( %ASCID 'MISCELLANEOUS')
236 0359 2 OR .FULL
237 0360 2 THEN
238 0361 2 BEGIN
239 0362 2 CHECK_INTER();
240 0363 2 DISPLAY_MISC();
241 0364 2 END;
242 0365 2 !
243 0366 2 IF CLISPRESNT( %ASCID 'RELOCATION')
244 0367 2 OR .FULL
245 0368 2 THEN
246 0369 2 BEGIN
247 0370 2 CHECK_INTER();
248 0371 2 DISPLAY_RELOC();
249 0372 2 END;
250 0373 2 !
251 0374 2 CHECK_INTER();
252 0375 2 !
253 0376 2 RETURN SSS_NORMAL;
254 0377 1 END;

```

```

.TITLE DISPIMDMP Display image dump data
.IDENT \V04-000\

```

```

.PSECT $SPLITS, NOWRT, NOEXE, 2

```

```

54 55 50 54 55 4F 24 53 59 53 00000 P.AAA: .ASCII \SYS$OUTPUT\
54 55 50 4E 49 24 53 59 53 0000A P.AAB: .ASCII \SYS$INPUT\
00013 .BLKB 1
00 00 54 55 50 54 55 4F 00014 P.AAD: .ASCII \OUTPUT\<0><0>
010E0006 0001C P.AAC: .LONG 17694726
00000000' 00020 .ADDRESS P.AAD
00 00 54 55 50 54 55 4F 00024 P.AAF: .ASCII \OUTPUT\<0><0>
010E0006 0002C P.AAE: .LONG 17694726
00000000' 00030 .ADDRESS P.AAF
4C 4C 55 46 00034 P.AAH: .ASCII \FULL\
010E0004 00038 P.AAG: .LONG 17694724

```



```

00 00 53 55 4F 45 4E 41 4C 4C 45 43 53 47 4D
00 00 4E 4F 49 54 41 43 4F 4C 45 52
00000000' 0003C
010E000D 00040 P.AAJ:
00000000' 0004F
00000000' 00050 P.AAI:
00000000' 00054
010E000A 00058 P.AAL:
00000000' 00064 P.AAK:
00000000' 00068

```

```

.PSECT $OWNS,NOEXE,2
.OUTFAB:

```

```

03 00000
50 00001
0000 00002
00000000 00004
00000000 00008
00000000 0000C
00000000 00010
0000 00014
03 00016
00 00017
00000000 00018
00 0001C
00 0001D
00 0001E
02 0001F
00000000 00020
00000000 00024
00000000 00028
00000000' 0002C
00000000 00030
0A 00034
00 00035
0000 00036
00000000 00038
0000 0003C
00 0003E
00 0003F
00000000 00040
00000000 00044
0000 00048
00 0004A
00 0004B
00000000 0004C
01 00050
44 00051
0000 00052
00000000 00054
00000000 00058
00000000 0005C
0000# 00060
0000 00066
00000000 00068
0000 0006C
00 0006E
00 0006F
0000 00070

```

```

.OUTFAB:
.BYTE 3
.BYTE 80
.WORD 0
.LONG 0
.LONG 0
.LONG 0
.LONG 0
.WORD 0
.BYTE 3
.BYTE 0
.LONG 0
.BYTE 0
.BYTE 0
.BYTE 2
.LONG 0
.LONG 0
.LONG 0
.ADDRESS P.AAA
.LONG 0
.BYTE 10
.BYTE 0
.WORD 0
.LONG 0
.WORD 0
.BYTE 0
.BYTE 0
.LONG 0
.LONG 0
.WORD 0
.BYTE 0
.BYTE 0
.LONG 0
.LONG 0
.WORD 0
.BYTE 1
.BYTE 68
.WORD 0
.LONG 0
.LONG 0
.WORD 0[3]
.WORD 0
.LONG 0
.WORD 0
.BYTE 0
.BYTE 0
.WORD 0

```

```

.OUTRAB:

```

.....

```

0000 00072
00000000 00074
00000000 00078
00000000 0007C
00000000 00080
00 00084
00 00085
00 00086
00 00087
00000000 00088
00000000 0008C
00000000 00090
0A 0D 00094
6F 74 20 4E 52 55 54 45 52 20 73 73 65 72 50 00096
20 3A 65 75 6E 69 74 6E 6F 63 20 000A5
000B0
03 000B4
50 000B5
0000 000B6
00000000 000B8
00000000 000BC
00000000 000C0
00000000 000C4
0000 000C8
03 000CA
00 000CB
00000000 000CC
00 000D0
00 000D1
00 000D2
02 000D3
00000000 000D4
00000000 000D8
00000000 000DC
00000000 000E0
00000000 000E4
09 000E8
00 000E9
0000 000EA
00000000 000EC
0000 000F0
00 000F2
00 000F3
00000000 000F4
00000000 000F8
0000 000FC
00 000FE
00 000FF
00000000 00100
01 00104
44 00105
0000 00106
40000000 00108
00000000 0010C
00000000 00110
0000# 00114
0000 0011A

```

```

.WORD 0
.LONG 0
.LONG 0
.LONG 0
.LONG 0
.BYTE 0
.BYTE 0
.BYTE 0
.BYTE 0
.LONG 0
.ADDRESS OUTFAB
.LONG 0
.PROMPT: .BYTE 13, 10
.ASCII \Press RETURN to continue: \
ANSWER: .BLKB 4
INFAB: .BYTE 3
.BYTE 80
.WORD 0
.LONG 0
.LONG 0
.LONG 0
.LONG 0
.WORD 0
.BYTE 3
.BYTE 0
.LONG 0
.BYTE 0
.BYTE 0
.BYTE 2
.LONG 0
.LONG 0
.LONG 0
.ADDRESS P.AAB
.LONG 0
.BYTE 9
.BYTE 0
.WORD 0
.LONG 0
.WORD 0
.BYTE 0
.BYTE 0
.LONG 0
.LONG 0
.WORD 0
.BYTE 0
.BYTE 0
.LONG 0
.BYTE 1
.INRAB: .BYTE 68
.WORD 0
.LONG 1073741824
.LONG 0
.LONG 0
.WORD 0[3]
.WORD 0

```

.....

```

00000000 0011C .LONG 0
      0000 00120 .WOFD 0
      00 00122 .BYTE 0
      00 00123 .BYTE 0
      0004 00124 .WORD 4
      0000 00126 .WORD 0
00000000 00128 .ADDRESS ANSWER
00000000 0012C .LONG 0
00000000 00130 .LONG 0
00000000 00134 .ADDRESS PROMPT
      1C 00138 .BYTE 28
      00 00139 .BYTE 0
      00 0013A .BYTE 0
      00 0013B .BYTE 0
00000000 0013C .LONG 0
00000000 00140 .ADDRESS INFAB
00000000 00144 .LONG 0
      00148 DSPBUF: .BLKB 512
      00348 DSPDESC: .LONG 512
      0034C .ADDRESS DSPBUF
      00350 DSPLN: .BLKB 4

```

```

.EXTRN IMGHDR, MISC, MISC VA
.EXTRN CLISGET VALUE, CLISPRESENT
.EXTRN SYSSCREATE, SYSSCONNECT
.EXTRN SYSSOPEN

```

```

.PSECT $CODE$,NOWRT,2

```

```

      007C 00000 .ENTRY DISPLAY DUMP, Save R2,R3,R4,R5,R6
56 0000V CF 9E 00002 MOVAB CHECK INTER, R6
55 00000000G 00 9E 00007 MOVAB SYSSCONNECT, R5
54 0000G CF 9E 0000E MOVAB CLISPRESENT, R4
53 0000' CF 9E 00013 MOVAB P.AAC, R3
5E 04 C2 00018 SUBL2 #4, SP
      53 DD 0001B PUSHL R3
64 01 FB 0001D CALLS #1, CLISPRESENT
20 50 E9 00020 BLBC R0, 1$
6E 020E0000 8F D0 00023 MOVL #34471936, OUTNAME
      6D D4 0002A CLRL OUTNAME+4
      5E DD 0002C PUSHL SP
      10 A3 9F 0002E PUSHAB P.AAE
      0000G CF 02 FB 00031 CALLS #2, CLISGET VALUE
      71 50 E9 00036 BLBC RETSTATUS, 8$
      0000' CF 6D D0 00039 MOVL OUTNAME+4, OUTFAB+44
      0000' CF 6E 90 0003E MOVB OUTNAME, OUTFAB+52
00000000G 00 0000' CF 9F 00043 1$: PUSHAB OUTFAB
59 01 FB 00047 CALLS #1, SYSSCREATE
      50 E9 0004E BLBC RETSTATUS, 6$
      0000' CF 9F 00051 PUSHAB OUTRAB
65 01 FB 00055 CALLS #1, SYSSCONNECT
4F 50 E9 00058 BLBC RETSTATUS, 6$
      0000' CF 9F 0005B PUSHAB INFAB
00000000G 00 01 FB 0005F CALLS #1, SYSSOPEN
      1 50 E9 00066 BLBC RETSTATUS, 6$
      0000' CF 9F 00069 PUSHAB INRAB
65 01 FB 0006D CALLS #1, SYSSCONNECT

```



```
256 0378 1 ROUTINE CHECK_INTER : NOVALUE =
257 0379 2 BEGIN
258 0380 2
259 0381 2 ++
260 0382 2
261 0383 2 Functional Description:
262 0384 2 This routine pauses for interactive use.
263 0385 2
264 0386 2 Calling Sequence:
265 0387 2 standard
266 0388 2
267 0389 2 Input Parameters:
268 0390 2 none
269 0391 2
270 0392 2 Implicit Inputs:
271 0393 2 command line
272 0394 2
273 0395 2 Output Parameters:
274 0396 2 none
275 0397 2
276 0398 2 Implicit Outputs:
277 0399 2 none
278 0400 2
279 0401 2 Routines Called:
280 0402 2 CLISPRESNT
281 0403 2
282 0404 2 Routine Value:
283 0405 2 none
284 0406 2
285 0407 2 Signals:
286 0408 2 none
287 0409 2
288 0410 2 Side Effects:
289 0411 2 none
290 0412 2
291 0413 2 --
292 0414 2
293 0415 2 OWN
294 0416 2 FIRST; ! Is this the first call
295 0417 2
296 0418 2
297 0419 2 IF CLISPRESNT( %ASCID 'INTERACTIVE')
298 0420 2 AND
299 0421 2 .FIRST NEQ 0
300 0422 2 THEN
301 0423 2 BEGIN
302 0424 2 $GET (RAB = INRAB);
303 0425 2
304 0426 2 ! Exit on a ^Z
305 0427 2 !
306 0428 2 IF .ANSWER[0] EQL %X'1A'
307 0429 2 THEN
308 0430 2 $EXIT();
309 0431 2 END;
310 0432 2
311 0433 2 FIRST = .FIRST + 1;
312 0434 2
```

: 313 0435 1 END;

```

                                .PSECT $PLITS,NOWRT,NOEXE,2
00 45 56 49 54 43 41 52 45 54 4E 49 0006C P.AAN: .ASCII \INTERACTIVE\<0>
                                010E000B, 00078 P.AAM: .LONG 17694731
                                00000000, 0007C .ADDRESS P.AAN
                                .PSECT $OWNS,NOEXE,2
                                00354 FIRST: .BLKB 4
                                .EXTRN SYSSGET, SYSSEXIT
                                .PSECT $CODES,NOWRT,2
                                0000 00000 CHECK_INTER:
                                .WORD Save nothing
                                PUSHAB P.AAM
                                CALLS #1, CLIPRESENT
                                BLBC R0, 1$
                                TSTL FIRST
                                BEQL 1$
                                PUSHAB INRAB
                                CALLS #1, SYSSGET
                                CMPB ANSWER, #26
                                BNEQ 1$
                                PUSHL #1
                                CALLS #1, SYSSEXIT
                                INCL FIRST
                                RET
                                0000' CF 9F 00002
                                0000G CF 21 01 FB 00006
                                0000' CF D5 0000B 50 E9 0000B
                                0000' CF 1B 13 00012 1B 13 00012
                                0000G 00 01 FB 00014 01 FB 00014
                                0000' CF 91 0001F 01 FB 00018
                                0000' CF 09 12 00024 09 12 00024
                                0000G 00 01 DD 00026 01 DD 00026
                                0000' CF 01 FB 00028 01 FB 00028
                                0000' CF D6 0002F 1$: 04 00033
                                04 00033
                                : 0378
                                : 0419
                                :
                                : 0421
                                :
                                : 0424
                                :
                                : 0428
                                :
                                : 0430
                                :
                                : 0433
                                : 0435

```

: Routine Size: 52 bytes, Routine Base: \$CODES + 00AB

```

315 0436 1 ROUTINE DISPLAY_MISC =
316 0437 BEGIN
317 0438
318 0439 ++
319 0440
320 0441 Functional Description:
321 0442 This routine display misc items of data.
322 0443
323 0444 Calling Sequence:
324 0445 standard
325 0446
326 0447 Input Parameters:
327 0448 none
328 0449
329 0450 Implicit Inputs:
330 0451 misc data block
331 0452
332 0453 Output Parameters:
333 0454 none
334 0455
335 0456 Implicit Outputs:
336 0457 none
337 0458
338 0459 Routines Called:
339 0460 none
340 0461
341 0462 Routine Value:
342 0463 system service return codes
343 0464
344 0465 Signals:
345 0466 none
346 0467
347 0468 Side Effects:
348 0469 none
349 0470
350 0471 --
351 0472 LOCAL
352 0473 IHI : REF BLOCK[,BYTE], ! image ID part of image header
353 0474 IHP : REF BLOCK[,BYTE], ! patch info part of image header
354 0475 RETSTATUS;
355 0476
356 0477 DISPLAY_FIELD ( );
357 P 0478 DISPLAY_FIELD ( STRING = ' R0 = !XL R1 = !XL R2 = !XL R3 = !XL',
358 P 0479 FIELD1 = .MISC[IMGDMP$L_R0],
359 P 0480 FIELD2 = .MISC[IMGDMP$L_R1],
360 P 0481 FIELD3 = .MISC[IMGDMP$L_R2],
361 0482 FIELD4 = .MISC[IMGDMP$L_R3] );
362 P 0483 DISPLAY_FIELD ( STRING = ' R4 = !XL R5 = !XL R6 = !XL R7 = !XL',
363 P 0484 FIELD1 = .MISC[IMGDMP$L_R4],
364 P 0485 FIELD2 = .MISC[IMGDMP$L_R5],
365 P 0486 FIELD3 = .MISC[IMGDMP$L_R6],
366 0487 FIELD4 = .MISC[IMGDMP$L_R7] );
367 P 0488 DISPLAY_FIELD ( STRING = ' R8 = !XL R9 = !XL R10 = !XL R11 = !XL',
368 P 0489 FIELD1 = .MISC[IMGDMP$L_R8],
369 P 0490 FIELD2 = .MISC[IMGDMP$L_R9],
370 P 0491 FIELD3 = .MISC[IMGDMP$L_R10],
371 0492 FIELD4 = .MISC[IMGDMP$L_R11] );

```

```

372 P 0493 2 DISPLAY_FIELD ( STRING = ' SP = !XL      AP = !XL      FP = !XL',
373 P 0494 2 FIELD1 = .MISC[IMGDMP$SL_SP],
374 P 0495 2 FIELD2 = .MISC[IMGDMP$SL_AP],
375 0496 2 FIELD3 = .MISC[IMGDMP$SL_FP] );
376 P 0497 2 DISPLAY_FIELD ( );
377 P 0498 2 DISPLAY_FIELD ( STRING = ' FREE PO VA !XL      FREE_P1_VA !XL',
378 P 0499 2 FIELD1 = .MISC[IMGDMP$SL_FREE_PO],
379 0500 2 FIELD2 = .MISC[IMGDMP$SL_FREE_P1] );
380 P 0501 2 DISPLAY_FIELD ( STRING = ' Active ASTs !XB      Enabled ASTs !XB',
381 P 0502 2 FIELD1 = .MISC[IMGDMP$SL_A$ACT],
382 0503 2 FIELD2 = .MISC[IMGDMP$SL_A$TEN] );
383 P 0504 2 DISPLAY_FIELD ( STRING = ' Current Privileges !XL !XL',
384 P 0505 2 FIELD1 = .(MISC[IMGDMP$SQ_CURPRIV]+4),
385 0506 2 FIELD2 = .(MISC[IMGDMP$SQ_CURPRIV]) );
386 P 0507 2 DISPLAY_FIELD ( STRING = ' Event Flags ?XL !XL',
387 P 0508 2 FIELD1 = .MISC[IMGDMP$SL_EFCU],
388 0509 2 FIELD2 = .MISC[IMGDMP$SL_EFCS] );
389 P 0510 2 DISPLAY_FIELD ( STRING = ' Buffered I/O count/limit !ZL/!ZL',
390 P 0511 2 FIELD1 = .MISC[IMGDMP$SL_BIOCNT],
391 0512 2 FIELD2 = .MISC[IMGDMP$SL_BIOLM] );
392 P 0513 2 DISPLAY_FIELD ( STRING = ' Direct I/O count/limit !ZL/!ZL',
393 P 0514 2 FIELD1 = .MISC[IMGDMP$SL_DIOCNT],
394 0515 2 FIELD2 = .MISC[IMGDMP$SL_DIOLM] );
395 P 0516 2 DISPLAY_FIELD ( STRING = ' File count/limit !ZL/!ZL',
396 P 0517 2 FIELD1 = .MISC[IMGDMP$SL_FILCNT],
397 0518 2 FIELD2 = .MISC[IMGDMP$SL_FILLM] );
398 P 0519 2 DISPLAY_FIELD ( STRING = ' Process count/limit !ZL/!ZL',
399 P 0520 2 FIELD1 = .MISC[IMGDMP$SL_PRCNT],
400 0521 2 FIELD2 = .MISC[IMGDMP$SL_PRCLM] );
401 P 0522 2 DISPLAY_FIELD ( STRING = ' Timer queue count/limit !ZL/!ZL',
402 P 0523 2 FIELD1 = .MISC[IMGDMP$SL_TQCNT],
403 0524 2 FIELD2 = .MISC[IMGDMP$SL_TQLM] );
404 P 0525 2 DISPLAY_FIELD ( STRING = ' AST count/limit !ZL/!ZL',
405 P 0526 2 FIELD1 = .MISC[IMGDMP$SL_ASTCNT],
406 0527 2 FIELD2 = .MISC[IMGDMP$SL_A$TLM] );
407 P 0528 2 DISPLAY_FIELD ( STRING = ' Enqueue count/limit !ZL/!ZL',
408 P 0529 2 FIELD1 = .MISC[IMGDMP$SL_ENQCNT],
409 0530 2 FIELD2 = .MISC[IMGDMP$SL_ENQLM] );
410 P 0531 2 DISPLAY_FIELD ( STRING = ' Buffered I/O total !ZL      Direct I/O total !ZL',
411 P 0532 2 FIELD1 = .MISC[IMGDMP$SL_BUFIO],
412 0533 2 FIELD2 = .MISC[IMGDMP$SL_DIRIO] );
413 0534 2 DISPLAY_FIELD ( );
414 0535 2 IHI = IMGHDR + .IMGHDR[IHDSW_IMGIDOFF];
415 P 0536 2 DISPLAY_FIELD ( STRING = ' Link Date !XD',
416 0537 2 FIELD1 = IHI[IHISQ_LINKTIME]);
417 0538 2 IF .IMGHDR[IHDSW_PATCHOFF] NEQ 0
418 0539 2 THEN
419 0540 2 BEGIN
420 0541 2 IHP = IMGHDR + .IMGHDR[IHDSW_PATCHOFF];
421 P 0542 2 ADD_FIELD ( STRING = ' Patch Date !XD',
422 0543 2 FIELD1 = IHP[IHPSQ_PATDATE] );
423 P 0544 2 DISPLAY_FIELD ( STRING = ' ECO Level !XL !XL !XL !XL',
424 P 0545 2 FIELD1 = .IHP[IHPSL_ECO1],
425 P 0546 2 FIELD2 = .IHP[IHPSL_ECO2],
426 P 0547 2 FIELD3 = .IHP[IHPSL_ECO3],
427 0548 2 FIELD4 = .IHP[IHPSL_ECO4] );
428 0549 2 END;

```


: 429
: 430
: 431
0550 2
0551 2 RETURN SSS_NORMAL;
0552 1 END;

```

.PSECT $PLITS,NOWRT,NOEXE,2
00 00 0A 0D 00080 P.AAP: .ASCII <13><10><0><0>
      010E0002 00084 P.AAO: .LONG 17694722
      00000000' 00088 .ADDRESS P.AAP
20 31 52 09 4C 58 21 20 3D 20 30 52 20 0A 0D 0008C P.AAR: .ASCII <13><10>\ R0 = !XL\<9>\R1 = !XL\<9>
21 20 3D 20 33 52 09 4C 58 21 20 58 21 20 3D 0009B
      00000000' 000A1 .ASCII \R2 = !XL\<9>\R3 = !XL\<0><0>
      00 00 4C 58 000B0
      010E0026 000B4 P.AAQ: .LONG 17694758
      00000000' 000B8 .ADDRESS P.AAR
20 35 52 09 4C 58 21 20 3D 20 34 52 20 0A 0D 000BC P.AAT: .ASCII <13><10>\ R4 = !XL\<9>\R5 = !XL\<9>
21 20 3D 20 37 52 09 4C 58 21 20 58 21 20 3D 000CB
      00000000' 000D1 .ASCII \R6 = !XL\<9>\R7 = !XL\<0><0>
      00 00 4C 58 000E0
      010E0026 000E4 P.AAS: .LONG 17694758
      00000000' 000E8 .ADDRESS P.AAT
20 39 52 09 4C 58 21 20 3D 20 38 52 20 0A 0D 000EC P.AAV: .ASCII <13><10>\ R8 = !XL\<9>\R9 = !XL\<9>
3D 20 31 31 52 09 4C 58 21 20 3D 20 30 31 52 00101
      00000000' 00110 .ASCII \R10 = !XL\<9>\R11 = !XL\
      4C 58 21 20 00110
      010E0028 00114 P.AAU: .LONG 17694760
      00000000' 00118 .ADDRESS P.AAV
20 50 41 09 4C 58 21 20 3D 20 50 53 20 0A 0D 0011C P.AAX: .ASCII <13><10>\ SP = !XL\<9>\AP = !XL\<9>
      00 00 4C 58 21 20 3D 20 20 50 46 0012B
      00000000' 00131 .ASCII \FP = !XL\<0><0>
      010E001E 0013C P.AAW: .LONG 17694750
      00000000' 00140 .ADDRESS P.AAX
      00 00 0A 0D 00144 P.AAZ: .ASCII <13><10><0><0>
      010E0002 00148 P.AAY: .LONG 17694722
      00000000' 0014C .ADDRESS P.AAZ
20 20 41 56 5F 30 50 5F 45 45 52 46 20 0A 0D 00150 P.ABB: .ASCII <13><10>\ FREE_PO_VA !XL\<9><9>
4C 58 21 20 20 41 56 5F 31 50 5F 45 45 52 46 0015F
      00000000' 00164 .ASCII \FREE_P1_VA !XL\<0>
      00 00 4C 58 21 20 3D 20 20 50 46 00173
      010E0023 00174 P.ABA: .LONG 17694755
      00000000' 00178 .ADDRESS P.ABB
20 73 54 53 41 20 65 76 69 74 63 41 20 0A 0D 0017C P.ABD: .ASCII <13><10>\ Active ASTs !XB\<9><9>
58 21 20 73 54 53 41 20 64 65 6C 62 58 21 20 0018B
      00000000' 00191 .ASCII \Enabled ASTs !XB\<0><0><0>
      00 00 61 6E 45 001A0
      010E0025 001A4 P.ABC: .LONG 17694757
      00000000' 001A8 .ADDRESS P.ABD
76 69 72 50 20 74 6E 65 72 72 75 43 20 0A 0D 001AC P.ABF: .ASCII <13><10>\ Current Privileges !XL !X\
58 21 20 20 4C 58 21 20 20 73 65 67 65 6C 69 001BB
      00000000' 001CA .ASCII \L\<0>
      00 00 00 4C 001CC P.ABE: .LONG 17694751
      010E001F 001D0 .ADDRESS P.ABF
20 73 67 61 6C 46 20 74 6E 65 76 45 20 0A 0D 001D4 P.ABH: .ASCII <13><10>\ Event Flags !XL !XL\
      00000000' 001E3
      4C 58 21 20 20 4C 58 21 20 001E3
      010E0018 001EC P.ABG: .LONG 17694744

```


FD22	C3		63	B0	000BF	MOVW	DSPLEN, OUTRAB+34
		FD00	C3	9F	000C4	PUSHAB	OUTRAB
	65		01	FB	000C8	CALLS	#1, SYSSPUT
	7F		50	E9	000CB	BLBC	RETSTATUS, 8\$
			7E	D4	000CE	CLRQ	-(SP)
	7E	2C	A6	7D	000D0	MOVQ	MISC+52, -(SP)
		28	A6	DD	000D4	PUSHL	MISC+48
		F8	A3	9F	000D7	PUSHAB	DSPDESC
			53	DD	000DA	PUSHL	R3
		0000'	CF	9F	000DC	PUSHAB	P.AAW
	64		07	FB	000E0	CALLS	#7, SYSSFAO
FD28	7C	FD00	50	E9	000E3	BLBC	RETSTATUS, 9\$
FD22	C3	FD00	C3	9E	000E6	MOVAB	DSPBUF, OUTRAB+40
			63	B0	000ED	MOVW	DSPLEN, OUTRAB+34
	65		C3	9F	000F2	PUSHAB	OUTRAB
	7C		01	FB	000F6	CALLS	#1, SYSSPUT
			50	E9	000F9	BLBC	RETSTATUS, 10\$
			7E	7C	000FC	CLRQ	-(SP)
			7E	7C	000FE	CLRQ	-(SP)
		F8	A3	9F	00100	PUSHAB	DSPDESC
			53	DD	00103	PUSHL	R3
		0000'	CF	9F	00105	PUSHAB	P.AAY
	64		07	FB	00109	CALLS	#7, SYSSFAO
FD28	69	FD00	50	E9	0010C	BLBC	RETSTATUS, 10\$
FD22	C3	FD00	C3	9E	0010F	MOVAB	DSPBUF, OUTRAB+40
			63	B0	00116	MOVW	DSPLEN, OUTRAB+34
	65		C3	9F	0011B	PUSHAB	OUTRAB
	6A		01	FB	0011F	CALLS	#1, SYSSPUT
			50	E9	00122	BLBC	RETSTATUS, 11\$
	7E	40	7E	7C	00125	CLRQ	-(SP)
		F8	A6	7D	00127	MOVQ	MISC+72, -(SP)
			A3	9F	0012B	PUSHAB	DSPDESC
			53	DD	0012E	PUSHL	R3
		0000'	CF	9F	00130	PUSHAB	P.ABA
	64		07	FB	00134	CALLS	#7, SYSSFAO
FD28	6B	FD00	50	E9	00137	BLBC	RETSTATUS, 12\$
FD22	C3	FD00	C3	9E	0013A	MOVAB	DSPBUF, OUTRAB+40
			63	B0	00141	MOVW	DSPLEN, OUTRAB+34
	65		C3	9F	00146	PUSHAB	OUTRAB
	6C		01	FB	0014A	CALLS	#1, SYSSPUT
			50	E9	0014D	BLBC	RETSTATUS, 13\$
	7E	48	7E	7C	00150	CLRQ	-(SP)
		F8	A6	7D	00152	MOVQ	MISC+80, -(SP)
			A3	9F	00156	PUSHAB	DSPDESC
			53	DD	00159	PUSHL	R3
		0000'	CF	9F	0015B	PUSHAB	P.ABC
	64		07	FB	0015F	CALLS	#7, SYSSFAO
FD28	6D	FD00	50	E9	00162	BLBC	RETSTATUS, 14\$
FD22	C3	FD00	C3	9E	00165	MOVAB	DSPBUF, OUTRAB+40
			63	B0	0016C	MOVW	DSPLEN, OUTRAB+34
	65		C3	9F	00171	PUSHAB	OUTRAB
	6C		01	FB	00175	CALLS	#1, SYSSPUT
			50	E9	00178	BLBC	RETSTATUS, 15\$
			7E	7C	0017B	CLRQ	-(SP)
		50	A6	DD	0017D	PUSHL	MISC+88
		54	A6	DD	00180	PUSHL	MISC+92
		F8	A3	9F	00183	PUSHAB	DSPDESC

0496
0497
0500
0503
0506

		0000'	53 DD 00186	PUSHL R3
			CF 9F 00188	PUSHAB P.ABE
	64		07 FB 0018C	CALLS #7, SYSSFAO
	6B		50 E9 0018F 11\$:	BLBC RETSTATUS, 16\$
FD28	C3	FD F8	C3 9E 00192	MOVAB DSPBUF, OUTRAB+40
FD22	C3		63 B0 00199	MOVW DSPLN, OUTRAB+34
		FD00	C3 9F 0019E	PUSHAB OUTRAB
	65		01 FB 001A2	CALLS #1, SYSSPUT
	6A		50 E9 001A5 12\$:	BLBC RETSTATUS, 17\$
			7E 7C 001A8	CLRQ -(SP)
		58	A6 JD 001AA	PUSHL MISC+96
		5C	A6 DD 001AD	PUSHL MISC+100
		F8	A3 9F 001B0	PUSHAB DSPDESC
			53 DD 001B3	PUSHL R3
		0000'	CF 9F 001B5	PUSHAB P.ABG
	64		07 FB 001B9	CALLS #7, SYSSFAO
	7F		50 E9 001BC 13\$:	BLBC RETSTATUS, 18\$
FD28	C3	FD F8	C3 9E 001BF	MOVAB DSPBUF, OUTRAB+40
FD22	C3		63 B0 001C6	MOVW DSPLN, OUTRAB+34
		FD00	C3 9F 001CB	PUSHAB OUTRAB
	65		01 FB 001CF	CALLS #1, SYSSPUT
	7F		50 E9 001D2 14\$:	BLBC RETSTATUS, 19\$
			7E 7C 001D5	CLRQ -(SP)
	7E	60	A6 7D 001D7	MOVQ MISC+104, -(SP)
		F8	A3 9F 001DB	PUSHAB DSPDESC
			53 DD 001DE	PUSHL R3
		0000'	CF 9F 001E0	PUSHAB P.ABI
	64		07 FB 001E4	CALLS #7, SYSSFAO
	6A		50 E9 001E7 15\$:	BLBC RETSTATUS, 19\$
FD28	C3	FD F8	C3 9E 001EA	MOVAB DSPBUF, OUTRAB+40
FD22	C3		63 B0 001F1	MOVW DSPLN, OUTRAB+34
		FD00	C3 9F 001F6	PUSHAB OUTRAB
	65		01 FB 001FA	CALLS #1, SYSSPUT
	6A		50 E9 001FD 16\$:	BLBC RETSTATUS, 20\$
			7E 7C 00200	CLRQ -(SP)
	7E	74	A6 7D 00202	MOVQ MISC+124, -(SP)
		F8	A3 9F 00206	PUSHAB DSPDESC
			53 DD 00209	PUSHL R3
		0000'	CF 9F 0020B	PUSHAB P.ABK
	64		07 FB 0020F	CALLS #7, SYSSFAO
	6B		50 E9 00212 17\$:	BLBC RETSTATUS, 21\$
FD28	C3	FD F8	C3 9E 00215	MOVAB DSPBUF, OUTRAB+40
FD22	C3		63 B0 0021C	MOVW DSPLN, OUTRAB+34
		FD00	C3 9F 00221	PUSHAB OUTRAB
	65		01 FB 00225	CALLS #1, SYSSPUT
	6B		50 E9 00228	BLBC RETSTATUS, 22\$
			7E 7C 0022B	CLRQ -(SP)
	7E	0080	C6 7D 0022D	MOVQ MISC+136, -(SP)
		F8	A3 9F 00232	PUSHAB DSPDESC
			53 DD 00235	PUSHL R3
		0000'	CF 9F 00237	PUSHAB P.ABM
	64		07 FB 0023B	CALLS #7, SYSSFAO
	6B		50 E9 0023E 18\$:	BLBC RETSTATUS, 23\$
FD28	C3	FD F8	C3 9E 00241	MOVAB DSPBUF, OUTRAB+40
FD22	C3		63 B0 00248	MOVW DSPLN, OUTRAB+34
		FD00	C3 9F 0024D	PUSHAB OUTRAB
	65		01 FB 00251	CALLS #1, SYSSPUT

0509

0512

0515

0518

	6B		50	E9	00254	19\$:	BLBC	RETSTATUS, 24\$	
			7E	7C	00257		CLRQ	-(SP)	0521
	7E	0088	C6	7D	00259		MOVQ	MISC+144, -(SP)	
		F8	A3	9F	0025E		PUSHAB	DSPDESC	
			S3	DD	00261		PUSHL	R3	
		0000'	CF	9F	00263		PUSHAB	P.ABO	
	64		07	FB	00267		CALLS	#7, SYSSFAO	
	6B		50	E9	0026A	20\$:	BLBC	RETSTATUS, 25\$	
FD28	C3	FDF8	C3	9E	0026D		MOVAB	DSPBUF, OUTRAB+40	
FD22	C3		63	B0	00274		MOVW	DSPLN, OUTRAB+34	
			C3	9F	00279		PUSHAB	OUTRAB	
	65		01	FB	0027D		CALLS	#1, SYSSPUT	
	6B		50	E9	00280	21\$:	BLBC	RETSTATUS, 26\$	
			7E	7C	00283		CLRQ	-(SP)	0524
	7E	0090	C6	7D	00285		MOVQ	MISC+152, -(SP)	
		F8	A3	9F	0028A		PUSHAB	DSPDESC	
			S3	DD	0028D		PUSHL	R3	
		0000'	CF	9F	0028F		PUSHAB	P.ABO	
	64		07	FB	00293		CALLS	#7, SYSSFAO	
	6B		50	E9	00296	22\$:	BLBC	RETSTATUS, 27\$	
FD28	C3	FDF8	C3	9E	00299		MOVAB	DSPBUF, OUTRAB+40	
FD22	C3		63	B0	002A0		MOVW	DSPLN, OUTRAB+34	
			C3	9F	002A5		PUSHAB	OUTRAB	
	65		01	FB	002A9		CALLS	#1, SYSSPUT	
	6C		50	E9	002AC	23\$:	BLBC	RETSTATUS, 28\$	
			7E	7C	002AF		CLRQ	-(SP)	0527
	7E	0098	C6	7D	002B1		MOVQ	MISC+160, -(SP)	
		F8	A3	9F	002B6		PUSHAB	DSPDESC	
			S3	DD	002B9		PUSHL	R3	
		0000'	CF	9F	002BB		PUSHAB	P.ABS	
	64		07	FB	002BF		CALLS	#7, SYSSFAO	
	7F		50	E9	002C2	24\$:	BLBC	RETSTATUS, 29\$	
FD28	C3	FDF8	C3	9E	002C5		MOVAB	DSPBUF, OUTRAB+40	
FD22	C3		63	B0	002CC		MOVW	DSPLN, OUTRAB+34	
			C3	9F	002D1		PUSHAB	OUTRAB	
	65		01	FB	002D5		CALLS	#1, SYSSPUT	
	7F		50	E9	002D8	25\$:	BLBC	RETSTATUS, 30\$	
			7E	7C	002DB		CLRQ	-(SP)	0530
	7E	00A0	C6	7D	002DD		MOVQ	MISC+168, -(SP)	
		F8	A3	9F	002E2		PUSHAB	DSPDESC	
			S3	DD	002E5		PUSHL	R3	
		0000'	CF	9F	002E7		PUSHAB	P.ABU	
	64		07	FB	002EB		CALLS	#7, SYSSFAO	
	69		50	E9	002EE	26\$:	BLBC	RETSTATUS, 30\$	
FD28	C3	FDF8	C3	9E	002F1		MOVAB	DSPBUF, OUTRAB+40	
FD22	C3		63	B0	002F8		MOVW	DSPLN, OUTRAB+34	
			C3	9F	002FD		PUSHAB	OUTRAB	
	65		01	FB	00301		CALLS	#1, SYSSPUT	
	73		50	E9	00304	27\$:	BLBC	RETSTATUS, 31\$	
			7E	7C	00307		CLRQ	-(SP)	0533
			7C	A6	DD	00309	PUSHL	MISC+132	
			68	A6	DD	0030C	PUSHL	MISC+112	
			F8	A3	9F	0030F	PUSHAB	DSPDESC	
			S3	DD	00312		PUSHL	R3	
		0000'	CF	9F	00314		PUSHAB	P.ABW	
	64		07	FB	00318		CALLS	#7, SYSSFAO	
	72		50	E9	0031B	28\$:	BLBC	RETSTATUS, 32\$	

DISPIMDMP
V04-000

Display image dump data

F 11
16-Sep-1984 01:42:19
14-Sep-1984 12:35:01

VAX-11 Bliss-32 V4.0-742
[IMGDMP.SRC]DISPIMDMP.B32;1

Page 22
(5)

65	FD00	C3	9F	003E9	PUSHAB	OUTRAB
03		01	FB	003ED	CALLS	#1, SYSSPUT
50		50	E9	003F0	BLBC	REFSTATUS, 37\$
		01	D0	003F3	MOVL	#1, R0
			04	003F6	RET	

:
:
:
: 0551
: 0552

: Routine Size: 1015 bytes, Routine Base: \$CODE\$ + 00DF

: 432 0553 1


```

434 0554 1 ROUTINE DISPLAY_RELOC =
435 0555 2 BEGIN
436 0556 2
437 0557 2 !++
438 0558 2
439 0559 2 Functional Description:
440 0560 2 This routine display relocation data
441 0561 2
442 0562 2 Calling Sequence:
443 0563 2 standard
444 0564 2
445 0565 2 Input Parameters:
446 0566 2 none
447 0567 2
448 0568 2 Imolicit Inputs:
449 0569 2 misc data block
450 0570 2
451 0571 2 Output Parameters:
452 0572 2 none
453 0573 2
454 0574 2 Implicit Outputs:
455 0575 2 none
456 0576 2
457 0577 2 Routines Called:
458 0578 2 none
459 0579 2
460 0580 2 Routine Value:
461 0581 2 system service codes
462 0582 2
463 0583 2 Signals:
464 0584 2 none
465 0585 2
466 0586 2 Side Effects:
467 0587 2 none
468 0588 2
469 0589 2 !--
470 0590 2
471 0591 2 LOCAL
472 0592 2 RETSTATUS;
473 0593 2
474 0594 2 DISPLAY_FIELD ();
475 P 0595 2 DISPLAY_FIELD ( STRING = ' Kernel stack !XL pages at !XL moved to !XL',
476 P 0596 2 FIELD1 = .MISC[IMGDMP$KSTK_SIZ],
477 P 0597 2 FIELD2 = .MISC[IMGDMP$KSTK_VA],
478 0598 2 FIELD3 = .MISC_VA[IMGMOV$L_KSTK] );
479 P 0599 2 DISPLAY_FIELD ( STRING = ' Exec stack !XL pages at !XL moved to !XL',
480 P 0600 2 FIELD1 = .MISC[IMGDMP$ESTK_SIZ],
481 P 0601 2 FIELD2 = .MISC[IMGDMP$ESTK_VA],
482 0602 2 FIELD3 = .MISC_VA[IMGMOV$L_ESTK] );
483 P 0603 2 DISPLAY_FIELD ( STRING = ' Vector page !XL page at !XL moved to !XL',
484 P 0604 2 FIELD1 = .MISC[IMGDMP$VECPAG_SIZ],
485 P 0605 2 FIELD2 = .MISC[IMGDMP$VECPAG_VA],
486 0606 2 FIELD3 = .MISC_VA[IMGMOV$L_VECPAG] );
487 P 0607 2 DISPLAY_FIELD ( STRING = ' PIO (RMS) area !XL pages at !XL moved to !XL',
488 P 0608 2 FIELD1 = .MISC[IMGDMP$PIO_SIZ],
489 P 0609 2 FIELD2 = .MISC[IMGDMP$PIO_VA],
490 0610 2 FIELD3 = .MISC_VA[IMGMOV$L_PIO] );

```

```

: 491 P 0611 2 DISPLAY_FIELD ( STRING = ' Image activator context !XL page at !XL moved to !XL',
: 492 P 0612 2 FIELD1 = .MISC[IMGDMP$L-IMGCTX-SIZ],
: 493 P 0613 2 FIELD2 = .MISC[IMGDMP$L-IMGCTX-VA],
: 494 P 0614 2 FIELD3 = .MISC_VA[IMGMOV$L-IMGCTX] );
: 495 P 0615 2 DISPLAY_FIELD ( STRING = ' User writeable context !XL pages at !XL moved to !XL',
: 496 P 0616 2 FIELD1 = .MISC[IMGDMP$L-USRCTX-SIZ],
: 497 P 0617 2 FIELD2 = .MISC[IMGDMP$L-USRCTX-VA],
: 498 P 0618 2 FIELD3 = .MISC_VA[IMGMOV$L-USRCTX] );
: 499 P 0619 2
: 500 P 0620 2 RETURN SSS_NORMAL;
: 501 P 0621 1 END;

```

Address	Hex	Hex	Hex	Hex	Hex	Hex	Hex	Hex	Hex	Hex	Hex	Label	Description						
	00	00	0A	0D	003D0	P.ACH:	.ASCII	<13><10><0><0>											
					010E0002	P.ACG:	.LONG	17694722											
					00000000		.ADDRESS	P.ACH											
6B	63	61	74	73	20	6C	65	6E	72	65	4B	20	0A	0D	003DC	P.ACJ:	.ASCII	<13><10>\ Kernel stack !XL pages at !\	
21	20	74	61	20	73	65	67	61	70	20	4C	58	21	20	003EB				
4C	58	21	20	6F	74	20	64	65	76	6F	6D	20	4C	58	003FA		.ASCII	\XL moved to !XL\<0><0><0>	
												00	00	00	00409				
															010E002D	0040C	P.ACI:	.LONG	17694765
															00000000	00410		.ADDRESS	P.ACJ
21	20	6B	63	61	74	73	20	63	65	78	45	20	0A	0D	00414	P.ACL:	.ASCII	<13><10>\ Exec stack !XL pages at !XL\	
4C	58	21	20	74	61	20	73	65	67	61	70	20	4C	58	00423				
															00432		.ASCII	\ moved to !XL\<0>	
															010E002B	00440	P.ACK:	.LONG	17694763
															00000000	00444		.ADDRESS	P.ACL
20	65	67	61	70	20	72	6F	74	63	65	56	20	0A	0D	00448	P.ACN:	.ASCII	<13><10>\ Vector page !XL page at !XL\	
4C	58	21	20	74	61	20	65	67	61	70	20	4C	58	21	00457				
															00466		.ASCII	\ moved to !XL\<0>	
															010E002B	00474	P.ACM:	.LONG	17694763
															00000000	00478		.ADDRESS	P.ACN
72	61	20	29	53	4D	52	28	20	4F	49	50	20	0A	0D	0047C	P.ACP:	.ASCII	<13><10>\ PIO (RMS) area !XL pages at\	
74	61	20	73	65	67	61	70	20	4C	58	21	20	61	65	0048B				
21	20	6F	74	20	64	65	76	6F	6D	20	4C	58	21	20	0049A		.ASCII	\ !XL moved to !XL\<0>	
															004A9				
															010E002F	004AC	P.ACO:	.LONG	17694767
															00000000	004B0		.ADDRESS	P.ACP
61	76	69	74	63	61	20	65	67	61	6D	49	20	0A	0D	004B4	P.ACR:	.ASCII	<13><10>\ Image activator context !XL\	
4C	58	21	20	74	78	65	74	6E	6F	63	20	72	6F	74	004C3				
6F	6D	20	4C	58	21	20	74	61	20	65	67	61	70	20	004D2		.ASCII	\ page at !XL moved to !XL\<0>	
															004E1				
															010E0037	004EC	P.ACQ:	.LONG	17694775
															00000000	004F0		.ADDRESS	P.ACR
62	61	65	74	69	72	77	20	72	65	73	55	20	0A	0D	004F4	P.ACT:	.ASCII	<13><10>\ User writeable context !XL \	
20	4C	58	21	20	74	78	65	74	6E	6F	63	20	65	6C	00503				
6F	6D	20	4C	58	21	20	74	61	20	73	65	67	61	70	00512		.ASCII	\pages at !XL moved to !XL\<0>	
															00521				
															010E0037	0052C	P.ACS:	.LONG	17694775
															00000000	00530		.ADDRESS	P.ACT

.PSECT \$SPLITS,NOWRT,NOEXE,2

.PSECT \$CODES,NOWRT,2

FD28	C2	FD28	C2	9E	000C0	MOVAB	DSPBUF, OUTRAB+40	
FD22	C2	FD22	C2	62	80	000C7	MOVW	DSPLN, OUTRAB+34
		FD00	C2	9F	000CC	PUSHAB	OUTRAB	
	64		01	FB	000D0	CALLS	#1, SYSSPUT	
	77		50	E9	000D3	BLBC	RETSTATUS, 8\$	
			7E	D4	000D6	CLRL	-(SP)	
		0C	A6	DD	000D8	PUSHL	MISC VA+12	
		24	A5	DD	000DB	PUSHL	MISC+216	
		28	A5	DD	000DE	PUSHL	MISC+220	
		F8	A2	9F	000E1	PUSHAB	DSPDESC	
			52	DD	000E4	PUSHL	R2	
		0000'	CF	9F	000E6	PUSHAB	P.ACO	
	63		07	FB	000EA	CALLS	#7, SYSSFAO	
	79		50	E9	000ED	BLBC	RETSTATUS, 9\$	
FD28	C2	FD28	C2	9E	000F0	MOVAB	DSPBUF, OUTRAB+40	
FD22	C2	FD22	C2	62	80	000F7	MOVW	DSPLN, OUTRAB+34
		FD00	C2	9F	000FC	PUSHAB	OUTRAB	
	64		01	FB	00100	CALLS	#1, SYSSPUT	
	63		50	E9	00103	BLBC	RETSTATUS, 9\$	
			7E	D4	00106	CLRL	-(SP)	
		10	A6	DD	00108	PUSHL	MISC VA+16	
		30	A5	DD	0010B	PUSHL	MISC+228	
		34	A5	DD	0010E	PUSHL	MISC+232	
		F8	A2	9F	00111	PUSHAB	DSPDESC	
			52	DD	00114	PUSHL	R2	
		0000'	CF	9F	00116	PUSHAB	P.ACO	
	63		07	FB	0011A	CALLS	#7, SYSSFAO	
	49		50	E9	0011D	BLBC	RETSTATUS, 9\$	
FD28	C2	FD28	C2	9E	00120	MOVAB	DSPBUF, OUTRAB+40	
FD22	C2	FD22	C2	62	80	00127	MOVW	DSPLN, OUTRAB+34
		FD00	C2	9F	0012C	PUSHAB	OUTRAB	
	64		01	FB	00130	CALLS	#1, SYSSPUT	
	33		50	E9	00133	BLBC	RETSTATUS, 9\$	
			7E	D4	00136	CLRL	-(SP)	
		14	A6	DD	00138	PUSHL	MISC VA+20	
		3C	A5	DD	0013B	PUSHL	MISC+240	
		40	A5	DD	0013E	PUSHL	MISC+244	
		F8	A2	9F	00141	PUSHAB	DSPDESC	
			52	DD	00144	PUSHL	R2	
		0000'	CF	9F	00146	PUSHAB	P.ACS	
	63		07	FB	0014A	CALLS	#7, SYSSFAO	
	19		50	E9	0014D	BLBC	RETSTATUS, 9\$	
FD28	C2	FD28	C2	9E	00150	MOVAB	DSPBUF, OUTRAB+40	
FD22	C2	FD22	C2	62	80	00157	MOVW	DSPLN, OUTRAB+34
		FD00	C2	9F	0015C	PUSHAB	OUTRAB	
	64		01	FB	00160	CALLS	#1, SYSSPUT	
	03		50	E9	00163	BLBC	RETSTATUS, 9\$	
	50		01	D0	00166	MOVL	#1, R0	
			04	00169	9\$:	RET		

.....
0610
.....
0614
.....
0618
.....
0620
0621

: Routine Size: 362 bytes, Routine Base: \$CODE\$ + 0406

```

: 502      0622 1
: 503      0623 1 END
: 504      0624 0 ELUDOM

```

PSECT SUMMARY

Name	Bytes	Attributes
\$PLITS	1332	NOVEC,NOWRT, RD ,NOEXE,NOSHR, LCL, REL, CON,NOPIC,ALIGN(2)
\$OWNS	856	NOVEC, WRT, RD ,NOEXE,NOSHR, LCL, REL, CON,NOPIC,ALIGN(2)
\$CODES	1600	NOVEC,NOWRT, RD , EXE,NOSHR, LCL, REL, CON,NOPIC,ALIGN(2)

Library Statistics

File	----- Symbols -----		Pages Mapped	Processing Time
	Total	Loaded Percent		
_\$255\$DUA28:[SYSLIB]LIB.L32;1	18619	52 0	1000	00:01.9

COMMAND QUALIFIERS

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

Size: 1600 code + 2188 data bytes
Run Time: 00:40.4
Elapsed Time: 01:22.4
Lines/CPU Min: 926
Lexemes/CPU-Min: 30644
Memory Used: 394 pages
Compilation Complete

