

BDV11-AA,
KDF11-BA, BB

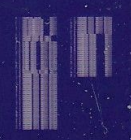
BDV11/KDF11B BT DIAG
CVM8ADO

AH-B062D-MC
FICHE 1 OF 1

OCT 1981
COPYRIGHT © 77 81
MADE IN USA



Grid of technical data tables with columns for parameters and values.



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44

002000

.ENABL ABS,AMA
.=2000
.NLIST CND,MD,MEB,ME
.TITLE USER DOCUMENTATION
.SBTTL IDENTIFICATION

; PRODUCT CODE: AC-B061D-MC

; PRODUCT NAME: CVM8ADO BDV11/KDF11B BT DIAG

; PRODUCT DATE: JUNE 1981

; MAINTAINER: DIAGNOSTIC ENGINEERING

:
: COPYRIGHT (C) 1977,1981
: DIGITAL EQUIPMENT CORPORATION, MAYNARD MASSACHUSSETTS 01754

: THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON A
: SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH THE INCLU-
: SION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE, OR ANY
: OTHER COPIES THEREOF, MAY NOT BE PROVIDED OR OTHERWISE MADE
: AVAILABLE TO ANY OTHER PERSON EXCEPT FOR USE ON SUCH SYSTEM
: AND TO ONE WHO AGREES TO THESE LICENSE TERMS. TITLE TO AND
: OWNERSHIP OF THE SOFTWARE SHALL AT ALL TIMES REMAIN IN DEC.

: THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT
: NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL
: EQUIPMENT CORPORATION.

: DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF
: ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DEC.
:

45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92

:++
: FUNCTIONAL DESCRIPTION:

THE BDV11 BOOTSTRAP/TERMINATOR/
DIAGNOSTIC MODULE PROVIDES THE
FOLLOWING FUNCTIONS:

1. ROM RESIDENT HARDWARE DIAGNOSTIC TESTS.
2. PADS FOR ROM RESIDENT BOOTSTRAP ROUTINES FOR THOSE DEVICES WHICH ARE SUPPORTED BY THE LSI-11 SYSTEM.
3. A READ/WRITE STORAGE REGISTER FOR USE BY THE RESIDENT DIAGNOSTIC TESTS.
4. TWELVE DIP ROCKER SWITCHES TO SELECT TESTING AND BOOTSTRAP OPTIONS AT POWER UP.
5. AN ARRAY OF FOUR LED'S TO PROVIDE STATUS INFORMATION.
6. HALT AND REBOOT TOGGLE SWITCHES FOR USE IN SYSTEMS WITHOUT A CONSOLE.
7. SOCKETS FOR 2K WORDS OF EPROM.
8. OPTIONAL REPLACEMENT OF SYSTEM ROM BY 8K WORDS OF EPROM.
9. LINE CLOCK INTERRUPT ENABLE/DISABLE REGISTER

:*BDV11 ONLY*

:*BDV11 ONLY*

:*BDV11 ONLY*

THE KDF11B BOOTSTRAP/DIAGNOSTIC MODULE PROVIDES THE FOLLOWING FUNCTIONS:

1. ABOVE MENTIONED FUNCTIONS 1., 3., 4.(8 SWITCHES), 5., 6., 9.
2. LINE CLOCK CLEARED BY RESET INSTRUCTION
3. PAGE CONTROL REGISTER IS WRITE ONLY
4. PRIORITY LEVEL 6(THE SAME FOR KDF11A)

:--

147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198

.SBTTL OPERATING INSTRUCTIONS

:1. LOADING AND STARTING PROCEDURES
: IN SYSTEMS OTHER THAN APT, THE DIAGNOSTIC PROGRAM
: AND THE DIAGNOSTIC SUPERVISOR WILL BE LOCATED ON THE XXDP+ MEDIA
: AS TWO SEPARATE FILES.

I. XXDP+ MEDIA

FOR OPERATING INSTRUCTIONS OF THE SUPERVISOR, PLEASE REFER
TO CHQUSB XXDP+/SUPR USER MAN (AC-F348 -MC).
ISSUE THE COMMAND ".R CVMBAD". THE XXDP+ MONITOR
WILL LOAD THE DIAGNOSTIC AND THE SUPERVISOR FILE
HSA A?? .SYS AND GIVE CONTROL TO THE SUPERVISOR.

II. SUPERVISOR COMMANDS

ONCE THE SUPERVISOR HAS BEEN INVOKED AT LOCATION 200,
THE FOLLOWING COMMANDS SHOULD BE USED SELECTIVELY TO
CONTROL THE RUNNING OF THE DIAGNOSTIC:

:2. TO START

START/TEST:<TESTNOS>/PASS:<PASSCNT>/UNIT:<DEVN>/FLAG:<CF>:<CF>

:WHERE:

TEST ::= (DEFINES WHICH TESTS TO EXECUTE, IF NO
SPECIFICATION EXECUTE ALL TESTS)
PASS ::= (INDICATES HOW MANY PASSES TO RUN, IF NO SPEC-
IFICATION RUN UNTIL DIAGNOSTIC ESCAPE SEQUENCE)
UNIT ::= (SPECIFIES WHICH UNIT ENTRIES TO GET FROM THE
CONFIGURATION FILE, IF NO SPECIFICATION USE ALL
APPLICABLE UNIT ENTRIES)
FLAG ::= (SPECIFIES THE ERROR CONTROL/REPORT FLAG OPTIONS
TO BE USED)
<TESTNOS> ::= (LIST FOR UP TO 16 TESTS TO BE EXECUTED IN AN
ASCENDING ORDER.)
<PASSCNT> ::= (NUMBER OF PROGRAM PASSES TO EXECUTE)
<DEVN> ::= (UNIQUE, DEC STANDARD, DEVICE SPECIFIER AND
UNIT NUMBER)
<CF> ::= (ANY OF THE FOLLOWING CONTROL FLAGS:
HOE-HALT ON ERROR
LOE-LOOP ON ERROR AND ATTEMPT REPORT
IER-INHIBIT ALL ERROR REPORTS
IBE-INHIBIT BASIC AND EXTENDED ERROR REPORTS
IXE-INHIBIT EXTENDED ERROR REPORTS
PRI-DIRECT ALL ERROR, PASS, AND STATISTICAL
REPORTS TO THE LINE PRINTER.
BOE-AUDIO ERROR INDICATION
UAM-UNATTENDED MODE, NO OPERATOR INTERVENTION
PNT-PRINT NUMBER OF TEST BEING EXECUTED.)

199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235

:3. TO RESTART
: THE RESTART COMMAND IS SIMILAR TO THE START COMMAND EXCEPT
: THAT ALL PARAMETERS ARE ASSUMED TO BE ALREADY DEFINED, AND NO
: OPERATOR DIALOGUE IS PERFORMED PRIOR TO RUNNING THE DIAGNOSTIC.
: IF THE OPERATOR WISHES TO ALTER THE TYPE OF ADDITIONAL MEMORY
: TO TEST, OR CHANGE THE ADDRESSES, LOCATION 'PASS' MUST BE
: CLEARED MANUALLY PRIOR TO RESTARTING, SINCE THIS INFORMATION
: IS SET UP ON THE FIRST PASS OF THE DIAGNOSTIC.
:
: RESTART/TEST:<TESTNOS>/PASS:<PASSCNT>/FLAG:<CF>:<CF>...
:
:4. TO RETURN TO PROGRAM
:
: TO RESUME EXECUTION OF THE DIAGNOSTIC AT THE FIRST INSTRUCTION
: FOLLOWING THE CURRENT SUPERVISOR CALL, AT WHICH TIME NEW FLAGS
: MAY BE ASSIGNED.
:
: CONTINUE/FLAG:<CF>:<CF>:...
:
:5. TO LOAD AND START THE DIAGNOSTIC
: TO LOAD AND START THE DIAGNOSTIC USING DEFAULT PARAMETERS
:
: RUN<FILESPEC>/TEST:<TESTNOS>/PASS:<PASSCNT>/UNIT:<DEVN>/FLAG:<CF>...
:
:NOTE: TEST NUMBERS AND UNIT NUMBERS MAY BE SPECIFIED
: AS SINGLE NUMBERS, RANGES OF NUMBERS (I.E. 1-6),
: OR COMBINATIONS OF BOTH.
:
: SPECIAL ENVIRONMENTS: APT
: TEST 7, THE TEST OF ALL RESIDENT MEMORY, WILL NOT RUN
: UNDER APT, AS IT REQUIRES USER INTERVENTION.
: THIS TEST DOES NOT RUN ON KDF11-B.
:
:***
: FOR MORE INFORMATION ON DRS FLAGS REFER TO XXDP+ USER'S
: MANUAL

236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277

: PROGRAM OPTIONS:
: THE HARDWARE PARAMETERS ARE STORED IN A PARAMETER TABLE WITH
: DEFAULT VALUES. THE OPERATOR WILL HAVE THE OPTION OF CHANGING
: THESE PARAMETERS BY RESPONDING TO THE APPROPRIATE QUESTIONS
: GENERATED BY THE DIAGNOSTIC SUPERVISOR. THESE PARAMETERS
: INCLUDE THE UNIT NUMBER, INTERRUPT VECTOR, PRIORITY LEVEL, AND
: ROCKER SWITCH SETTINGS. THE DEFAULT VALUES WILL BE TYPED ALONG
: WITH THE QUESTIONS.
:
: THERE ARE 12 SWITCHES ON BDV11 AND ONLY 8 ON KDF11-B.
: THE ROCKER SWITCH SETTINGS ARE EXAMINED IN THE FOLLOWING
: ORDER:
: B4 B3 B2 B1 A8 A7 A6 A5 A4 A3 A2 A1
: FOR EXAMPLE, IF SWITCHES A1, A2, A6, AND B1 WERE ON, THE SWITCH
: SETTING WOULD BE:
: B4 B3 B2 B1 A8 A7 A6 A5 A4 A3 A2 A1
: 1 1 1 1
: WHICH HAS AN OCTAL VALUE OF 0443.
:
: THE SOFTWARE P-TABLE CONTAINS THE CHECKWORDS FOR THE 2K
: OF DIAGNOSTIC ROM WHICH IS RESIDENT ON THE BDV11A. TO CHANGE
: THESE CHECKWORDS, THE OPERATOR MUST RESPOND WITH A YES TO THE
: SUPERVISOR'S QUESTION "CHANGE SW (Y/N)?". THE DEFAULT VALUES WILL
: THEN BE PRINTED AS THE QUESTIONS ARE ASKED.
:
: TEST 7(FOR BDV11 ONLY) CHECKS ALL THE ADDITIONAL MEMORY THAT
: IT IS INSTRUCTED TO TEST. THIS TEST IS SET UP BY THE OPERATOR ON THE
: FIRST PASS OF THE DIAGNOSTIC. THE DIAGNOSTIC WILL ASK IF THERE
: IS ANY ADDITIONAL MEMORY TO TEST, AND IF SO WILL ASK WHICH
: TYPE OF MEMORY IT IS. (THE OPERATOR CAN ANSWER THESE QUESTIONS
: WITH LOGICAL Y/N ANSWERS.) IF ANY ADDITIONAL MEMORY IS TO BE
: TESTED, THE OPERATOR MUST SUPPLY THE CHECKWORDS FOR THOSE
: ROMS/EPROMS. IN THE CASE OF SYSTEM ROM/EPROM, THE OPERATOR WILL
: ALSO HAVE TO INDICATE HOW MANY CHECKWORDS WILL BE INPUT (IN DECIMAL).
: NOTE THAT ONCE THIS DATA IS SET UP, THIS MEMORY WILL ALWAYS BE
: TESTED, EVEN IF THE DIAGNOSTIC IS RESTARTED, UNLESS THE LOCATION
: "PASS" IS CLEARED (SEE SEC.3 OF LOADING AND STARTING PROCEDURES).
:
: EXECUTION TIMES: A SINGLE ERROR-FREE PASS WILL REQUIRE
: LESS THAN 1 SEC. TO RUN UNDER APT. WHEN RUN
: IN STAND-ALONE MODE, IT WILL REQUIRE LESS
: THAN 3 SECS. TO RUN.
:

278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297

.SBTTL ERROR INFORMATION

: ERROR REPORTING PROCEDURES:
: IN GENERAL, ALL ERROR REPORTS WILL CONTAIN THE FOLLOWING
: INFORMATION:
: 1. A HEADER OF TEST IDENTIFICATION INFORMATION.
: THIS INCLUDES THE PROGRAM NAME, TYPE OF ERROR,
: ERROR NUMBER, TEST AND SUBTEST NUMBERS, UNIT
: NUMBER, AND AN OPTIONAL ADDITIONAL MESSAGE.
: 2. BASIC ERROR INFORMATION.
: THIS IS A SPECIFIC STATEMENT OF WHAT THE ERROR
: IS AND WHICH REGISTER OR ROM WAS INVOLVED.
: 3. EXTENDED ERROR INFORMATION.
: THIS IS OPTIONAL INFORMATION WHICH IS USED
: PRIMARILY TO GIVE THE EXPECTED AND ACTUAL
: CONTENTS OF THE APPROPRIATE DEVICE REGISTER
: DURING REGISTER TESTS.

	.SBTTL SUBTEST SUMMARIES		
	TEST NO.	SUBTEST NO.	PURPOSE
298	1	1	TO VERIFY THAT THE READ/WRITE REGISTER CAN HOLD ALL ZEROES.
299		2	TO VERIFY THAT THE READ/WRITE REGISTER CAN HOLD ALL ONES.
300		3	TO VERIFY THAT THE READ/WRITE REGISTER CAN HOLD AN ALTERNATING 1'S AND 0'S BIT PATTERN.
301		4	TO VERIFY THAT THE READ/WRITE REGISTER IS BYTE ADDRESSABLE.
302		5	TO VERIFY THAT THE READ/WRITE REGISTER CAN SWAP BYTES.
303		6	TO VERIFY THAT THE READ/WRITE REGISTER CAN HOLD AN ALTERNATING 0' AND 1'S BIT PATTERN.
304		7	TO VERIFY THAT THE READ/WRITE REGISTER IS BYTE ADDRESSABLE.
305		8	TO VERIFY THAT THE READ/WRITE REGISTER CAN SWAP BYTES.
306		9	TO VERIFY THAT THE READ/WRITE REGISTER CAN ROTATE A SET BIT WITHOUT PICKING UP ANY BITS.
307		10	TO VERIFY THAT THE READ/WRITE REGISTER CAN ROTATE A CLEAR BIT WITHOUT PICKING UP ANY BITS.
308	2 *BDV11 ONLY*	1	TEST 2 IS THE SAME AS TEST 1 EXCEPT THAT THE PAGE CONTROL REGISTER IS THE REGISTER UNDER TEST.
309		2	SAME AS TEST 1.
310		3	SAME AS TEST 1.
311		4	SAME AS TEST 1.
312		5	SAME AS TEST 1.
313		6	SAME AS TEST 1.
314		7	SAME AS TEST 1.
315		8	SAME AS TEST 1.
316		9	SAME AS TEST 1.
317		10	SAME AS TEST 1.
318	3	1	TO VERIFY THAT THE BEVENT CLAMP DISABLE ALLOWS INTERRUPTS WHEN OFF.
319		2	TO VERIFY THAT THE BEVENT CLAMP DISABLE INHIBITS INTERRUPTS WHEN ON.
320		3	TO VERIFY THAT PRIORITY 5 ALLOWS INTERRUPTS IF PRIORITY OF A DEVICE IS 6 (KDF11-A,KDF11-B)
321		4	TO VERIFY THAT PRIORITY 6 DOESN'T ALLOW INTERRUPTS IF PRIORITY OF A DEVICE IS 6 (KDF11-A,KDF11-B)
322		5	TO VERIFY THAT RESET WORKS FOR KDF11-B
323			
324			
325			
326			
327			
328			
329			
330			
331			
332			
333			
334			
335			
336			
337			
338			
339			
340			
341			
342			
343			
344			
345			
346			
347			
348			
349			
350			
351			
352			
353			

354	:	6	1	TO VERIFY THAT THE LOW BYTE
355	:			DIAGNOSTIC ROM HAS GOOD DATA.
356	:		2	TO VERIFY THAT THE HIGH BYTE
357	:			DIAGNOSTIC ROM HAS GOOD DATA.
358	:		3	TO INSURE THAT THE DIAGNOSTIC
359	:		4	ROMS HAVE NOT BEEN INTERCHANGED.
360	:	7	1	TO DETERMINE IF THERE IS ANY
361	:	*BDV11 ONLY*		ADDITIONAL MEMORY TO TEST.
362	:			THIS INFORMATION IS OBTAINED
363	:			THROUGH USER DIALOGUE.
364	:		2	TO TEST THE EXPANDED DIAGNOSTIC
365	:			ROM. FIRST THE REQUIRED CHECK-
366	:			WORDS MUST BE INPUT, AND THE
367	:			STARTING LOCATION IN MEMORY.
368	:			CHECKSUMS AND CHECKWORD
369	:			VERIFICATION CONFIRMS GOOD
370	:			DATA IN ROMS.
371	:		3	TO TEST THE EPROM IN THE
372	:			SOCKETS. TEST PROCEDURE IS AS
373	:			IN SUBTEST 2.
374	:		4	TO TEST SYSTEM ROM. SAME
375	:			TEST PROCEDURE AS IN SUBTEST 2.
376	:		5	TO TEST SYSTEM EPROM. SAME
377	:			TEST PROCEDURE AS IN SUBTEST 2.

```

378
379 002000          SVC
380                SVCINS=0
381                SVCGBL=0
382                SVCTAG=0
383                .TITLE PROGRAM HEADER AND TABLES
384                .SBTTL IDENTIFICATION
385
386
387                .SBTTL PROGRAM HEADER
388
389 002000          BGNMOD MDHEDR
390 002000          MDHEDR::
391
392                :++
393                : THE PROGRAM HEADER IS THE INTERFACE BETWEEN
394                : THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
395                :--
396
397 002000          POINTER BGNSW,BGNSFT
398
399
400 002000          HEADER CVM8A,D,0,7,0,340
401 002000          L$NAME:: :DIAGNOSTIC NAME
402 002000          103      .ASCII /C/
403 002001          126      .ASCII /V/
404 002002          115      .ASCII /M/
405 002003          070      .ASCII /8/
406 002004          101      .ASCII /A/
407 002005          000      .BYTE 0
408 002006          000      .BYTE 0
409 002007          000      .BYTE 0
410 002010          L$REV:: :REVISION LEVEL
411 002010          104      .ASCII /D/
412 002011          L$DEPO:: :0
413 002011          060      .ASCII /0/
414 002012          L$UNIT:: :NUMBER OF UNITS
415 002012          000000   .WORD 0
416 002014          L$TIML:: :LONGEST TEST TIME
417 002014          000007   .WORD 7
418 002016          L$HPCP:: :POINTER TO H.W. QUES.
419 002016          016566   .WORD L$HARD
420 002020          L$SPCP:: :POINTER TO S.W. QUES.
421 002020          017006   .WORD L$SOFT
422 002022          L$HPTP:: :PTR. TO DEF. H.W. TABLE
423 002022          002144   .WORD L$HW
424 002024          L$SPTP:: :PTR. TO S.W. TABLE
425 002024          002160   .WORD L$SW
426 002026          L$LADP:: :DIAG. END ADDRESS
427 002026          017024   .WORD L$LAST
428 002030          L$STA:: :RESERVED FOR APT STATS
429 002030          000000   .WORD 0
430 002032          L$CO:: :
431 002032          000000   .WORD 0
432 002034          L$DTYP:: :DIAGNOSTIC TYPE
433 002034          000000   .WORD 0
  
```

434	002036		L\$APT::		;APT EXPANSION
435	002036	000000		.WORD 0	
436	002040		L\$DTP::		;PTR. TO DISPATCH TABLE
437	002040	002124		.WORD L\$DISPATCH	
438	002042		L\$PRIO::		;DIAGNOSTIC RUN PRIORITY
439	002042	000340		.WORD 340	
440	002044		L\$ENVI::		;FLAGS DESCRIBE HOW IT WAS SETUP
441	002044	000000		.WORD 0	
442	002046		L\$EXP1::		;EXPANSION WORD
443	002046	000000		.WORD 0	
444	002050		L\$MREV::		;SVC REV AND EDIT #
445	002050	003		.BYTE C\$REVISION	
446	002051	003		.BYTE C\$EDIT	
447	002052		L\$EF::		;DIAG. EVENT FLAGS
448	002052	000000		.WORD 0	
449	002054	000000		.WORD 0	
450	002056		L\$SPC::		
451	002056	000000		.WORD 0	
452	002060		L\$DEVP::		; POINTER TO DEVICE TYPE LIST
453	002060	003156		.WORD L\$DVTYP	
454	002062		L\$REPP::		;PTR. TO REPORT CODE
455	002062	000000		.WORD 0	
456	002064		L\$EXP4::		
457	002064	000000		.WORD 0	
458	002066		I\$EXP5::		
459	002066	000000		.WORD 0	
460	002070		L\$AUT::		;PTR. TO ADD UNIT CODE
461	002070	000000		.WORD 0	
462	002072		L\$DUT::		;PTR. TO DROP UNIT CODE
463	002072	000000		.WORD 0	
464	002074		L\$LUN::		;LUN FOR EXERCISERS TO FILL
465	002074	000000		.WORD 0	
466	002076		L\$DESP::		;PTR. TO DIAG. DESCRIPTION
467	002076	003104		.WORD L\$DESC	
468	002100		L\$LOAD::		;GENERATE SPECIAL AUTOLOAD EMT
469	002100	104035		EMT E\$LOAD	
470	002102		L\$ETP::		;PTR. TO ERR_TBL
471	002102	000000		.WORD 0	
472	002104		L\$ICP::		;PTR. TO INIT CODE
473	002104	005252		.WORD L\$INIT	
474	002106		L\$CCP::		;PTR. TO CLEAN-UP CODE
475	002106	005332		.WORD L\$CLEAN	
476	002110		L\$ACP::		;PTR. TO AUTO CODE
477	002110	005330		.WORD L\$AUTO	
478	002112		L\$PRT::		;PTR. TO PROTECT TABLE
479	002112	005322		.WORD L\$PROT	
480	002114		L\$TEST::		;TEST NUMBER
481	002114	000000		.WORD 0	
482	002116		L\$DLY::		;DELAY COUNT
483	002116	000000		.WORD 0	
484	002120		L\$HIME::		;PTR. TO HIGH MEM
485	002120	000000		.WORD 0	
486	002122			ENDMOD	
487					

```
488 .SBTTL DISPATCH TABLE
489
490 :++
491 : THE DISPATCH TABLE CONTAINS THE STARTING ADDRESS OF EACH TEST.
492 : IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.
493 :--
494
495 002122          BGNMOD  DSPCODE
496 002122          DSPCODE::
497 002122          DISPATCH 7
498 002122 000007  .WORD 7
499 002124          L$DISPATCH::
500 002124 005412  .WORD T1
501 002126 006256  .WORD T2
502 002130 007144  .WORD T3
503 002132 010540  .WORD T4
504 002134 010710  .WORD T5
505 002136 011542  .WORD T6
506 002140 013646  .WORD T7
507 002142          ENDMOD
508
509 .SBTTL DEFAULT HARDWARE P-TABLE
510
511 :++
512 : THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF
513 : THE TEST-DEVICE PARAMETERS. THE STRUCTURE OF THIS TABLE
514 : IS IDENTICAL TO THE STRUCTURE OF THE RUN-TIME P-TABLE.
515 :--
516
517 002142          BGNHW  DFPTBL
518 002142 000005  .WORD L10000-L$HW/2
519 002144          L$HW::
520 002144          DFPTBL::
521
522          ;DEFAULT VALUES FOR UP TO SIX UNITS
523 002144 000000  .WORD 0          ;UNIT NUMBER 0
524 002146 000100  .WORD 100        ;INTERRUPT VECTOR
525 002150 000007  .WORD 7          ;PRIORITY LEVEL
526 002152 007777  .WORD 7777       ;ROCKER SWITCH SETTINGS
527 002154 000000  .WORD 0          ;BDV11 = 0, KDF11-B = 1
528
529 002156          ENDDHW
530 002156          L10000:
531
532 .SBTTL SOFTWARE P-TABLE
533
534 :++
535 : THE SOFTWARE P-TABLE CONTAINS THE VALUES OF THE PROGRAM
536 : PARAMETERS THAT CAN BE CHANGED BY THE OPERATOR.
537 :--
538
539 002156          BGNSW  SFPTBL
540 002156 000030  .WORD L10001-L$SW/2
541 002160          L$SW::
542 002160          SFPTBL::
543
```

```
544
545
546
547
548
549 002160 017042
550 002162 020656
551 002164 065162
552 002166 161744
553 002170 124453
554 002172 113667
555 002174 056040
556 002176 044734
557
558
559 002200 031547
560 002202 014036
561 002204 065162
562 002206 124632
563 002210 032040
564 002212 167124
565 002214 155461
566 002216 032257
567
568
569 002220 166014
570 002222 166015
571 002224 040621
572 002226 010700
573 002230 066675
574 002232 163042
575 002234 171300
576 002236 077220
577
578 002240
579 002240
580
581
```

:THE SOFTWARE P-TABLE IS USED TO STORE THE CHECKWORDS
:FOR THE DIAGNOSTIC ROM WHICH IS TESTED IN TEST 6.

:THE CHECKWORDS CORRESPONDING TO ROM CHIPS #23-045E2 AND #23-046E2 FOLLOW:

.WORD	17042	:ROMA: PAGE	0,1
.WORD	20656	:ROMB: PAGE	2,3
.WORD	65162	:ROMC: PAGE	4,5
.WORD	161744	:ROMD: PAGE	6,7
.WORD	124453	:ROME: PAGE	10,11
.WORD	113667	:ROMF: PAGE	12,13
.WORD	56040	:ROMG: PAGE	14,15
.WORD	44734	:ROMH: PAGE	16,17

:THE CHECKWORDS CORRESPONDING TO ROM CHIPS #23-010E2 AND #23-011E2 FOLLOW:

.WORD	31547	:ROMA: PAGE	0,1
.WORD	14036	:ROMB: PAGE	2,3
.WORD	65162	:ROMC: PAGE	4,5
.WORD	124632	:ROMD: PAGE	6,7
.WORD	32040	:ROME: PAGE	10,11
.WORD	167124	:ROMF: PAGE	12,13
.WORD	155461	:ROMG: PAGE	14,15
.WORD	32257	:ROMH: PAGE	16,17

:THE CHECKWORDS CORRESPONDING TO ROM CHIPS #23-244E2 AND #23-245E2 FOLLOW:

.WORD	166014	:ROMA: PAGE	0,1
.WORD	166015	:ROMB: PAGE	2,3
.WORD	040621	:ROMC: PAGE	4,5
.WORD	010700	:ROMD: PAGE	6,7
.WORD	066675	:ROME: PAGE	10,11
.WORD	163042	:ROMF: PAGE	12,13
.WORD	171300	:ROMG: PAGE	14,15
.WORD	077220	:ROMH: PAGE	16,17

ENDSW
L10001:

```

582          .TITLE GLOBAL AREAS
583          .SBTTL IDENTIFICATION
584
585
586          .SBTTL GLOBAL EQUATES SECTION
587
588          :++
589          : THE GLOBAL EQUATES SECTION CONTAINS PROGRAM EQUATES THAT
590          : ARE USED IN MORE THAN ONE TEST.
591          :--
592
593          BGNMOD GLBEQAT
594          GLBEQAT::
595          EQUALS
596
597          ; BIT DIFINITIONS
598          ;
599          100000 BIT15== 100000
600          040000 BIT14== 40000
601          020000 BIT13== 20000
602          010000 BIT12== 10000
603          004000 BIT11== 4000
604          002000 BIT10== 2000
605          001000 BIT09== 1000
606          000400 BIT08== 400
607          000200 BIT07== 200
608          000100 BIT06== 100
609          000040 BIT05== 40
610          000020 BIT04== 20
611          000010 BIT03== 10
612          000004 BIT02== 4
613          000002 BIT01== 2
614          000001 BIT00== 1
615
616          BIT9== BIT09
617          BIT8== BIT08
618          BIT7== BIT07
619          BIT6== BIT06
620          BIT5== BIT05
621          BIT4== BIT04
622          BIT3== BIT03
623          BIT2== BIT02
624          BIT1== BIT01
625          BIT0== BIT00
626
627          ; EVENT FLAG DEFINITIONS
628          ; EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
629
630          000040 EF.START== 32. ; START COMMAND WAS ISSUED
631          000037 EF.RESTART== 31. ; RESTART COMMAND WAS ISSUED
632          000036 EF.CONTINUE== 30. ; CONTINUE COMMAND WAS ISSUED
633          000035 EF.NEW== 29. ; A NEW PASS HAS BEEN STARTED
634          000034 EF.PWR== 28. ; A POWER-FAIL/POWER-UP OCCURRED
635
636          ;
637          ; PRIORITY LEVEL DEFINITIONS

```

```
638  
639      000340  
640      000300  
641      000240  
642      000200  
643      000140  
644      000100  
645      000040  
646      000000  
647  
648  
649  
650      000004  
651      000010  
652      000020  
653      000040  
654      000100  
655      000200  
656      000400  
657      001000  
658      002000  
659      004000  
660      010000  
661      020000  
662      040000  
663      100000  
664  
665      177520  
666      177524  
667 002240
```

```
;  
PRI07== 340  
PRI06== 300  
PRI05== 240  
PRI04== 200  
PRI03== 140  
PRI02== 100  
PRI01== 40  
PRI00== 0  
;  
;OPERATOR FLAG BITS  
;  
EVL==      4  
LOT==     10  
ADR==     20  
IDU==     40  
ISR==    100  
UAM==    200  
BOE==    400  
PNT==   1000  
PRI==   2000  
IXE==   4000  
IBE==  10000  
IER==  20000  
IOE==  40000  
HOE== 100000  
  
PCR=177520  
LSREG=177524  
ENDMOD
```


668
669
670
671
672
673
674
675 002240
676 002240
677 002240 000000
678 002242 000000
679 002244 000000
680 002246 000000
681 002250 000000
682 002252 000000
683 002254 000001
684 002256 000000
685 002260 000000
686 002262 000000
687 002264 000000
688 002266 000000
689 002270 000000
690 002272 000000
691 002274 000100
692 002276 000000
693 002300 000000
694 002302 000000
695 002304 000000
696 002306 000000
697 002310 000000
698 002312 000000
699 002314 000001
700 002316 000000
701 002320 000000
702 002322 000000
703 002324 000000
704 002326 000000
705 002330 000010
706 002350 000010
707 002370 000100
708 002570
709
710
711

.SBTTL GLOBAL DATA SECTION

;++
: THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
: IN MORE THAN ONE TEST.
:--

BGNMOD	GLBDAT
GLBDAT::	
KDF11B:	.WORD 0
VRTPCR:	.WORD 0
BCF:	.WORD 0
REAL:	.WORD 0
LOPAG:	.WORD 0
COUNTR:	.WORD 0
ANSR:	.WORD 1
RFLAG:	.WORD 0
EXPSUM:	.WORD 0
ACTSUM:	.WORD 0
PASS:	.WORD 0
PASCT:	.WORD 0
ULIMIT:	.WORD 0
PAGE:	.WORD 0
VECT:	.WORD 100
SWSET:	.WORD 0
STORE:	.WORD 0
WORDCT:	.WORD 0
PRIOR:	.WORD 0
CKWD:	.WORD 0
BADWD:	.WORD 0
RESPND:	.WORD 0
RSET:	.WORD 1
LORANG:	.WORD 0
HIRANG:	.WORD 0
BYTLOC:	.WORD 0
ERRFLG:	.WORD 0
DELCNT:	.WORD 0
EXPDIA:	.BLKW 10
EPROM:	.BLKW 10
SYSROM:	.BLKW 100
ENDMOD	

:=1 IF RUNNING ON A KDF11-B
:VIRTUAL PAGE CONTROL REGISTER

:EXPANDED DIAG. ROM CHECKWORDS
:EPROM CHECKWORDS
:SYSTEM ROM/EPROM CHECKWORDS

712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767

.SBTTL GLOBAL TEXT SECTION

:++
: THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
: MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
: MORE THAN ONE TEST.
:--

:GLOBAL MESSAGES

RWR: .ASCIZ 'READ/WRITE REGISTER ADDRESS: 177522'

PACR: .ASCIZ /PAGE CONTROL REGISTER ADDRESS: 177520/

CKERR: .ASCIZ /CHECKSUM ERROR/

CWDERR: .ASCIZ /INCORRECT CHECKWORD/

LOBYT: .ASCIZ /ERROR OCCURRED IN A LOW BYTE PAGE/

HIBYT: .ASCIZ /ERROR OCCURRED IN A HIGH BYTE PAGE/

LOADR: .ASCIZ /START OF MEMORY RANGE (K)/

.EVEN

```

768
769
770
771      : NAMES OF DEVICES SUPPORTED BY PROGRAM
772      :
773      :   DESCRIPT      <CVMBAD BDV11\KDF11-B BOOTSTRAP DIAGNOSTIC>
774      :   L$DESC::
775      :   .ASCIZ  /CVMBAD BDV11\KDF11-B BOOTSTRAP DIAGNOSTIC/
776      :
777      :
778      :
779      :
780      :
781      :
782      :   .EVEN
783      :   DEVTYP  <BDV11\KDF11-B>
784      :   L$DVTYP::
785      :   .ASCIZ  /BDV11\KDF11-B/
786      :
787      :
788      :   .EVEN
789
790
791      :
792      :   : FORMAT STATEMENTS USED IN PRINT CALLS
793      :
794
795
796      :   ZERR:  .ASCIZ  /%REGISTER CANNOT HOLD ALL ZEROES%/
797
798
799
800
801
802
803      :   ONERR: .ASCIZ  /%REGISTER CANNOT HOLD ALL ONES%/
804
805
806
807
808
809
810      :   BDDAT: .ASCIZ  /%REGISTER CANNOT HOLD GOOD DATA%/
811
812
813
814
815
816
817      :   BYTINS: .ASCIZ  /%REGISTER IS NOT BYTE ADDRESSABLE%/
818
819
820
821
822
823

```

773	003104			
774	003104			
775	003104	053103	034115	042101
776	003112	041040	053104	030461
777	003120	045534	043104	461
778	003126	041055	041040	7517
779	003134	051524	051124	050101
780	003142	042040	040511	047107
781	003150	051517	044524	000103
783	003156			
784	003156			
785	003156	042102	030526	056061
786	003164	042113	030506	026461
787	003172	000102		
796	003174	040445	042522	044507
797	003202	052123	051105	041440
798	003210	047101	047516	020124
799	003216	047510	042114	040440
800	003224	046114	055040	051105
801	003232	042517	022523	000116
803	003240	040445	042522	044507
804	003246	052123	051105	041440
805	003254	047101	047516	020124
806	003262	047510	042114	040440
807	003270	046114	047440	042516
808	003276	022523	000116	
810	003302	040445	042522	044507
811	003310	052123	051105	041440
812	003316	047101	047516	020124
813	003324	047510	042114	043440
814	003332	047517	020104	040504
815	003340	040524	047045	000
817	003345	045	051101	043505
818	003352	051511	042524	020122
819	003360	051511	047040	052117
820	003366	041040	052131	020105
821	003374	042101	051104	051505
822	003402	040523	046102	022505
823	003410	000116		

824
825 003412 040445 042522 044507 ROT1: .ASCIZ /%AREGISTER PICKED UP AN EXTRA SET BIT%N/
826 003420 052123 051105 050040
827 003426 041511 042513 020104
828 003434 050125 040440 020116
829 003442 054105 051124 020101
830 003450 042523 020124 044502
831 003456 022524 000116
832
833 003462 040445 042522 044507 ROTO: .ASCIZ /%AREGISTER PICKED UP AN EXTRA CLEAR BIT%N/
834 003470 052123 051105 050040
835 003476 041511 042513 020104
836 003504 050125 040440 020116
837 003512 054105 051124 020101
838 003520 046103 040505 020122
839 003526 044502 022524 000116
840
841 003534 040445 047125 041101 DIAGER: .ASCIZ /%AUNABLE TO LOCATE CORRECT MEMORY PAGE%N/
842 003542 042514 052040 020117
843 003550 047514 040503 042524
844 003556 041440 051117 042522
845 003564 052103 046440 046505
846 003572 051117 020131 040520
847 003600 042507 047045 000
848
849 003605 045 046501 046505 VIRMSG: .ASCIZ /%AMEMORY RANGE: %D2%A - %D2%AK%N/
850 003612 051117 020131 040522
851 003620 043516 035105 022440
852 003626 031104 040445 026440
853 003634 022440 031104 040445
854 003642 022513 000116
855
856 003646 040445 054105 042520 REGDT: .ASCIZ /%AEXPECTED: %06%S5%ARECEIVED: %06%N/
857 003654 052103 042105 020072
858 003662 047445 022466 032523
859 003670 040445 042522 042503
860 003676 053111 042105 020072
861 003704 047445 022466 000116
862
863 .EVEN

864
865
866
867
868
869
870
871
872
873
874 003712
875 003712
876 003712
877 003712 012746 003174
878 003716 012746 000001
879 003722 010600
880 003724 104414
881 003726 062706 000004
882 003732
883 003732 010246
884 003734 010146
885 003736 012746 003646
886 003742 012746 000003
887 003746 010600
888 003750 104415
889 003752 062706 000010
890 003756
891 003756
892 003756 104423
893
894 003760
895 003760
896 003760
897 003760 012746 003240
898 003764 012746 000001
899 003770 010600
900 003772 104414
901 003774 062706 000004
902 004000
903 004000 010246
904 004002 010146
905 004004 012746 003646
906 004010 012746 000003
907 004014 010600
908 004016 104415
909 004020 062706 000010
910 004024
911 004024
912 004024 104423
913
914 004026
915 004026
916 004026
917 004026 012746 003302
918 004032 012746 000001
919 004036 010600

.SBTTL GLOBAL ERROR REPORT SECTION

```

:++
: THE GLOBAL ERROR REPORT SECTION CONTAINS THE PRINTB AND PRINTX CALLS
: THAT ARE USED IN MORE THAN ONE TEST. IT ALSO INCLUDES THE ASCII MESSAGES
: THAT ARE USED BY THE PRINTB AND PRINTX CALLS.
:--
    
```

```

BGNMSG RERR1
RERR1::
PRINTB #ZERR
      MOV #ZERR,-(SP)
      MOV #1,-(SP)
      MOV SP,R0
      TRAP C$PNTB
      ADD #4,SP
PRINTX #REGDT,R1,R2
      MOV R2,-(SP)
      MOV R1,-(SP)
      MOV #REGDT,-(SP)
      MOV #3,-(SP)
      MOV SP,R0
      TRAP C$PNTX
      ADD #10,SP
    
```

```

ENDMSG
L10002:
      TRAP C$MSG
    
```

```

BGNMSG RERR2
RERR2::
PRINTB #ONERR
      MOV #ONERR,-(SP)
      MOV #1,-(SP)
      MOV SP,R0
      TRAP C$PNTB
      ADD #4,SP
PRINTX #REGDT,R1,R2
      MOV R2,-(SP)
      MOV R1,-(SP)
      MOV #REGDT,-(SP)
      MOV #3,-(SP)
      MOV SP,R0
      TRAP C$PNTX
      ADD #10,SP
    
```

```

ENDMSG
L10003:
      TRAP C$MSG
    
```

```

BGNMSG RERR3
RERR3::
PRINTB #BDDAT
      MOV #BDDAT,-(SP)
      MOV #1,-(SP)
      MOV SP,R0
    
```

920	004040	104414			TRAP	C\$PNTB
921	004042	062706	000004		ADD	#4,SP
922	004046			PRINTX	#REGDT,R1,R2	
923	004046	010246			MOV	R2,-(SP)
924	004050	010146			MOV	R1,-(SP)
925	004052	012746	003646		MOV	#REGDT,-(SP)
926	004056	012746	000003		MOV	#3,-(SP)
927	004062	010600			MOV	SP,R0
928	004064	104415			TRAP	C\$PNTX
929	004066	062706	000010		ADD	#10,SP
930	004072			ENDMSG		
931	004072			L10004:		
932	004072	104423			TRAP	C\$MSG
933						
934	004074			BGNMSG	RERR4	
935	004074			RERR4::		
936	004074			PRINTB	#BYTINS	
937	004074	012746	003345		MOV	#BYTINS,-(SP)
938	004100	012746	000001		MOV	#1,-(SP)
939	004104	010600			MOV	SP,R0
940	004106	104414			TRAP	C\$PNTB
941	004110	062706	000004		ADD	#4,SP
942	004114			PRINTX	#REGDT,R1,R2	
943	004114	010246			MOV	R2,-(SP)
944	004116	010146			MOV	R1,-(SP)
945	004120	012746	003646		MOV	#REGDT,-(SP)
946	004124	012746	000003		MOV	#3,-(SP)
947	004130	010600			MOV	SP,R0
948	004132	104415			TRAP	C\$PNTX
949	004134	062706	000010		ADD	#10,SP
950	004140			ENDMSG		
951	004140			L10005:		
952	004140	104423			TRAP	C\$MSG
953						
954	004142			BGNMSG	RERR5	
955	004142			RERR5::		
956	004142			PRINTB	#ROT1	
957	004142	012746	003412		MOV	#ROT1,-(SP)
958	004146	012746	000001		MOV	#1,-(SP)
959	004152	010600			MOV	SP,R0
960	004154	104414			TRAP	C\$PNTB
961	004156	062706	000004		ADD	#4,SP
962	004162			ENDMSG		
963	004162			L10006:		
964	004162	104423			TRAP	C\$MSG
965						
966	004164			BGNMSG	RERR6	
967	004164			RERR6::		
968	004164			PRINTB	#ROTO	
969	004164	012746	003462		MOV	#ROTO,-(SP)
970	004170	012746	000001		MOV	#1,-(SP)
971	004174	010600			MOV	SP,R0
972	004176	104414			TRAP	C\$PNTB
973	004200	062706	000004		ADD	#4,SP
974	004204			ENDMSG		
975	004204			L10007:		

976	004204	104423		TRAP	C\$MSG
977					
978	004206			BGNMSG	PAGERR
979	004206			PAGERR::	
980	004206			PRINTB	#DIAGER
981	004206	012746	003534	MOV	#DIAGER,-(SP)
982	004212	012746	000001	MOV	#1,-(SP)
983	004216	010600		MOV	SP,R0
984	004220	104414		TRAP	C\$PNTB
985	004222	062706	000004	ADD	#4,SP
986	004226			ENDMSG	
987	004226			L10010:	
988	004226	104423		TRAP	C\$MSG
989					
990					
991					
992	004230			VIPRI:	PRINTF #VIRMSG,LORANG,HIRANG
993	004230	013746	002320	MOV	HIRANG,-(SP)
994	004234	013746	002316	MOV	LORANG,-(SP)
995	004240	012746	003605	MOV	#VIRMSG,-(SP)
996	004244	012746	000003	MOV	#3,-(SP)
997	004250	010600		MOV	SP,R0
998	004252	104417		TRAP	C\$PNTF
999	004254	062706	000010	ADD	#10,SP
1000					
1001				.EVEN	
1002					
1003					
1004					
1005					
1006					

1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023 004260 012701 173776
1024 004264 063701 002244
1025 004270 005037 002262
1026 004274 012702 173000
1027 004300 063702 002244
1028 004304 111204
1029 004306 060437 002262
1030 004312 062702 000002
1031 004316 020201
1032 004320 002771
1033 004322 000207
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043 004324
1044 004324 012746 004402
1045 004330 012746 000001
1046 004334 010600
1047 004336 104417
1048 004340 062706 000004
1049 004344
1050 004344 104443
1051 004346 000406
1052 004350 002300
1053 004352 000022
1054 004354 004470
1055 004356 177777
1056 004360 000000
1057 004362 177777
1058 004364
1059 004364 013722 002300
1060 004370 005337 002302
1061 004374 001401
1062 004376 000762

.SBTTL GLOBAL SUBROUTINES SECTION

;++
: THE GLOBAL SUBROUTINES SECTION CONTAINS THE SUBROUTINES
: THAT ARE USED IN MORE THAN ONE TEST.
:--

;++
: FUNCTIONAL DESCRIPTION:
: SUBROUTINE TO COMPUTE A CHECKSUM IN A ROM/EPROM
: INPUT: CONTENTS OF BCF
: IMPLICIT INPUTS: CONTENTS OF PCR
: OUTPUT: A CHECKSUM VALUE STORED IN LOCATION ACTSUM
: CALLING SEQUENCE: JSR PC,CHKSUM
:--

```
CHKSUM: MOV      #173776,R1      ;STORE THE HIGHEST ADDRESS IN THE ROM
        ADD      BCF,R1        ;FOR EITHER LOW OR HIGH BYTES
        CLR      ACTSUM       ;CLEAR LOCATION WHICH WILL HOLD THE CHECKSUM
        MOV      #173000,R2    ;COMPUTE THE LOWEST ADDRESS IN THE ROM
        ADD      BCF,R2        ;WHERE THE DATA WILL START
1$:     MOV      (R2),R4        ;GET DATA IN BYTES
        ADD      R4,ACTSUM     ;ADD CONTENTS OF EACH LOCATION TO THE CHECKSUM
        ADD      #2,R2        ;ADJUST ADDRESS
        CMP      R2,R1        ;COMPARE CURRENT ADDRESS WITH HIGHEST ADDRESS
        BLT     1$           ;BR IF LESS THAN
        RTS      PC          ;RETURN
```

;++
: SUBROUTINE TO INPUT CHECKWORDS FROM THE OPERATOR
: INPUTS: NUMBER OF CHECKWORDS TO INPUT
: POINTER TO STORAGE AREA
: OUTPUTS: CHECKWORDS STORED IN PROPER TABLE
: CALLING SEQUENCE: JSR PC,INPUT
:--

```
INPUT:  PRINTF  #INSTR          ;PRINT INSTRUCTIONS
        MOV     #INSTR,-(SP)
        MOV     #1,-(SP)
        MOV     SP,R0
        TRAP   C$PNTF
        ADD     #4,SP
INLP:   GMANID  INWORD,STORE,0,-1,0,177777,NO
        TRAP   C$GMAN
        BR     10000$
        .WORD  STORE
        .WORD  T$CODE
        .WORD  INWORD
        .WORD  -1
        .WORD  T$LLOLIM
        .WORD  T$HILIM
10000$: MOV     STORE,(R2)+      ;PUT CHECKWORD IN TABLE
        DEC     WORDCT        ;DECREMENT WORD COUNT
        BEQ    1$           ;BR IF FINISHED
        BR     INLP         ;LOOP UNTIL TABLE IS COMPLETE
```



```
1063 004400 000207          1$:      RTS      PC          ;RETURN
1064
1065 004402 040445 054524 042520 INSTR:  .ASCIZ  /%ATYPE IN THE CHECKWORDS AS LISTED IN THE PRINT SET%N/
1066 004410 044440 020116 044124
1067 004416 020105 044103 041505
1068 004424 053513 051117 051504
1069 004432 040440 020123 044514
1070 004440 052123 042105 044440
1071 004446 020116 044124 020105
1072 004454 051120 047111 020124
1073 004462 042523 022524 000116
1074
1075 004470 044103 041505 053513 INWORD: .ASCIZ /CHECKWORD: /
1076 004476 051117 035104 000040
1077
1078          .EVEN
1079
```

```

1080      ;++
1081      ;SUBROUTINE TO COMPUTE THE VIRTUAL ADDRESS OF A BAD
1082      ;PAGE IN MEMORY
1083      ;INPUTS: PAGE IN PAGE CONTROL REGISTER
1084      ;          BYTE CONTROL FLAG (BCF)
1085      ;OUTPUTS: MEMORY RANGE IN WHICH ERROR OCCURRED
1086      ;CALLING SEQUENCE: JSR PC,VIRTAD
1087      ;--
1088
1089      004504 005001      VIRTAD: CLR      R1          ;START AT BOTTOM OF RANGE
1090      004506 012737 000007 002270      MOV      #7,ULIMIT      ;SET UPPER LIMIT OF PAGE
1091      004514 113737 177520 002272      MOVB     PCR,PAGE      ;LOW PAGE ERROR
1092      004522 023737 002272 002270      LPADD:  CMP      PAGE,ULIMIT ;IS PAGE <=ULIMIT
1093      004530 003430      BLE      OUTPUT      ;BR IF YES
1094      004532 022737 000057 002270      CMP      #57,ULIMIT    ;IS ULIMIT = 57
1095      004540 001006      BNE      1$          ;BR IF NO
1096      004542 012737 000207 002270      MOV      #207,ULIMIT   ;CHANGE UPPER LIMIT
1097      004550 012701 000020      MOV      #20,R1        ;ADJUST MEMORY POINTER
1098      004554 000762      BR       LPADD        ;CHECK PAGE AGAIN
1099      004556 062737 000010 002270      1$:    ADD      #10,ULIMIT ;INCREASE UPPER LIMIT
1100      004564 022737 000377 002270      CMP      #377,ULIMIT  ;HAS THE UPPER LIMIT EXCEEDED THE MAX. PAGE
1101      004572 002004      BGE      2$          ;BR IF NO
1102      004574      ERRDF  40,,PAGERR ;COULD NOT FIND THE PAGE OF MEMORY
1103      004574 104455      TRAP    C$ERDF
1104      004576 000050      .WORD  40
1105      004600 000000      .WORD  0
1106      004602 004206      .WORD  PAGERR
1107      004604      2$:    CKLOOP
1108      004604 104406      TRAP    C$CLP1
1109      004606 005201      INC     R1          ;ADJUST POINTER
1110      004610 000744      BR      LPADD        ;LOOP UNTIL UPPER LIMIT IS FOUND
1111      004612 010137 002316      OUTPUT: MOV     R1,LORANG ;PULL THE LOW RANGE OUT OF THE TABLE
1112      004616 013737 002316 002320      MOV     LORANG,HIRANG ;COPY THE DATA
1113      004624 005237 002320      INC     HIRANG      ;INCREMENT TO OBTAIN 1K RANGE
1114      004630 005737 002256      TST     RFLAG      ;IS IT ROM (2K SEGMENTS)
1115      004634 001402      BEQ     3$          ;BR IF NO
1116      004636 005237 002320      INC     HIRANG      ;OBTAIN 2K RANGE
1117      004642 000207      3$:    RTS     PC      ;RETURN
1118

```

```

1119      :++
1120      :SUBROUTINE TO VERIFY THE CHECKSUM VALUE OF A PAGE
1121      :OF EXISTENT MEMORY AND ALSO TEST FOR THE PROPER CHECKWORD.
1122      :INPUTS: PAGE CONTROL REGISTER, PAGE CHECKWORD.
1123      :OUTPUTS: ERROR FLAGS WHICH POINT TO THE PROPER ERROR MESSAGE
1124      :SUBORDINATE ROUTINES USED: CHKSUM
1125      :CALLING SEQUENCE: JSR PC, MEMTST
1126      :--
1127
1128 004644 005037 002246      MEMTST: CLR      REAL      :CLEAR MEMORY INDICATOR
1129 004650 005037 002244      LOBYTE: CLR      BCF      :SIGNAL LOW BYTES ARE BEING CHECKED
1130 004654 122737 177777 173774      CMPB     #-1,@#173774 :DOES THE ROM EXIST
1131 004662 001421      BEQ      HIBYTE   :BR IF NO
1132 004664 005237 002246      INC      REAL      :INDICATE THAT MEMORY EXISTS
1133 004670 004737 004260      JSR      PC,CHKSUM :COMPUTE THE ACTUAL CHECKSUM
1134 004674 113737 173776 002260      MOVB    @#173776,EXPSUM :GET THE STORED CHECKSUM
1135 004702 063737 002262 002260      ADD     ACTSUM,EXPSUM :ADD THE EXPECTED AND ACTUAL CHECKSUMS
1136 004710 105737 002260      TSTB    EXPSUM     :TEST RESULTING CHECKBYTE
1137 004714 001404      BEQ     1$        :BR IF NO ERROR
1138 004716 012737 000001 002324      MOV     #1,ERRFLG :SET CHECKSUM ERROR FLAG
1139 004724 000207      RTS     PC        :RETURN
1140 004726      1$:
1141
1142 004726 012737 000001 002244      HIBYTE: MOV     #1,BCF      :SET BCF TO DENOTE HIGH BYTES
1143 004734 122737 177777 173775      CMPB    #-1,@#173775 :DOES THE ROM EXIST
1144 004742 001427      BEQ     TSTCKW   :BR IF NO
1145 004744 005737 002246      TST     REAL     :WAS THERE A LOW ROM?
1146 004750 001003      BNE     2$        :BR IF YES
1147 004752 005037 002246      CLR     REAL     :DENOTE NON-EXISTENT LOW ROM
1148 004756 000207      RTS     PC        :RETURN FOR ERROR MESSAGE
1149 004760 005237 002246      2$:  INC     REAL     :INDICATE MEMORY EXISTS
1150 004764 004737 004260      JSR     PC,CHKSUM :COMPUTE CHECKSUM
1151 004770 113737 173777 002260      MOVB    @#173777,EXPSUM :GET EXPECTED CHECKSUM
1152 004776 063737 002262 002260      ADD     ACTSUM,EXPSUM :ADD THE EXPECTED AND ACTUAL CHECKSUMS
1153 005004 105737 002260      TSTB    EXPSUM   :TEST RESULTING CHECKBYTE
1154 005010 001404      BEQ     TSTCKW   :BR IF EQUAL
1155 005012 012737 000001 002324      MOV     #1,ERRFLG :SET CHECKSUM ERROR FLAG
1156 005020 000207      RTS     PC        :RETURN
1157
1158 005022 005737 002246      TSTCKW: TST     REAL     :ANY MEMORY?
1159 005026 001434      BEQ     5$        :BR IF NO
1160 005030 022737 000001 002246      CMP     #1,REAL   :SINGLE ROM?
1161 005036 001016      BNE     3$        :BR IF NO
1162 005040 123737 002306 173776      CMPB    CKWD,@#173776 :COMPARE CHECKBYTE ONLY
1163 005046 001001      BNE     100$     :BR IF ERROR
1164 005050 000207      RTS     PC        :RETURN -- NO ERROR
1165 005052 005037 002310      100$: CLR     BADWD    :CLEAR LOCATION
1166 005056 012737 000002 002324      MOV     #2,ERRFLG :DENOTE CHECKSUM ERROR
1167 005064 113737 173776 002310      MOVB    @#173776,BADWD :STORE BAD BYTE
1168 005072 000207      RTS     PC        :RETURN
1169 005074 023737 002306 173776      3$:  CMP     CKWD,@#173776 :COMPARE CHECKWORD
1170 005102 001404      BEQ     5$        :BR IF NO ERROR
1171 005104 012737 000002 002324      4$:  MOV     #2,ERRFLG :DENOTE CHECKSUM ERROR
1172 005112 013737 173776 002310      MOV     @#173776,BADWD :STORE WRONG CHECKWORD
1173 005120 000207      5$:  RTS     PC        :RETURN
1174

```

```

1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186 005122 013701 002300
1187 005126 020127 000005
1188 005132 003006
1189 005134 000241
1190 005136 006101
1191 005140 006101
1192 005142 006101
1193 005144 110104
1194 005146 000413
1195 005150 012704 000020
1196 005154 012705 000200
1197 005160 020104
1198 005162 001404
1199 005164 005204
1200 005166 062705 000010
1201 005172 000772
1202 005174 010504
1203 005176 110437 002250
1204 005202 005204
1205 005204 110437 002251
1206 005210 000207
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222 005212 010146
1223 005214 010246
1224 005216 012502
1225 005220 012737 000502 002326
1226
1227 005226 013701 002326
1228 005232 005301
1229 005234 001376
1230 005236 005302

:++
:SUBROUTINE TO COMPUTE THE ACTUAL STARTING PAGE
:OF MEMORY IN WHICH THE MEMORY CHIP IS TO BE
:ADDRESSED.
:INPUTS: THE LOW NUMBER IN THE MEMORY RANGE
:        (I.E. X IN X-Y K)
:OUTPUT: PAGE NUMBER IN PCR WHICH DENOTES WHERE TESTING
:        SHOULD BEGIN.
:CALLING SEQUENCE: JSR PC,SETADR
:--

SETADR: MOV     STORE,R1           ;COPY DATA
        CMP     R1,#5           ;IS THE NUMBER <=5?
        BGT     1$             ;BR IF NO
        CLC                    ;CLEAR C-BIT FOR ROTATE
        ROL     R1             ;ROTATE TO MULTIPLY
        ROL     R1             ; BY 10 (8)
        ROL     R1             ;
        MOVB    R1,R4          ;COPY DATA
        BR      LOAD           ;LOAD THE PCR
1$:     MOV     #20,R4          ;START WITH 16 (10)
        MOV     #200,R5        ;CORRESPONDIGE PAGE IS 200
LOOP:   CMP     R1,R4          ;PAGE FOUND?
        BEQ     2$             ;BR IF YES
        INC     R4             ;NEXT PAGE
        ADD     #10,R5         ;NEXT PAGE
        BR      LOOP          ;LOOP UNTIL PAGE IS FOUND
2$:     MOV     R5,R4          ;GET PAGE FOR PCR
LOAD:   MOVB    R4,LOPAG       ;LOW STARTING PAGE
        INC     R4             ;INCREMENT
        MOVB    R4,LOPAG+1     ;HIGH STARTING PAGE
        RTS     PC

:++
:SUBROUTINE TO DELAY IN MSECS
:DELAY IS USED IN TWO TESTS
:TEST 3 - BEVENT CLAMP ENABLE TEST
:TEST 4 - LIGHT DISPLAY TEST
:TIMING LOOP IS NOT CRITICAL FOR TESTS,SO THE SAME
:TIMER LOOP IS USED FOR LSI-11 AND 11/23.
:LOOP WILL BE 2.5 TIMES SLOWER FOR LSI-11
:
:CALL
:      JSR     R5,WDELAY       ;40 MSECS
:      40.
:
WDELAY: MOV     R1,-(SP)
        MOV     R2,-(SP)
        MOV     (R5)+,R2      ;APPROX. MSEC DELAY
1$:     MOV     #322,DELCNT    ;11/23 APPROX. 1 MSEC LOOP
        ;10% TOLERANCE
2$:     MOV     DELCNT,R1
3$:     DEC     R1             ;START APPROX. 1 MSEC LOOP
        BNE    3$
        DEC     R2             ;CHECK ON MSECS REQUESTED

```

GLOBAL AREAS
CVMBAD.P11

MACY11 30(1046)
10-JUL-81 17:08

20-JUL-81 11:44 PAGE 28

C 3

GLOBAL SUBROUTINES SECTION

SEQ 0028

1231 005240 001372
1232 005242 012602
1233 005244 012601
1234 005246 000205
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250

BNE 2\$
MOV (SP)+,R2
MOV (SP)+,R1
RTS R5

:BRANCH AND DO ANOTHER MSEC
:SETUP FOR RETURN AFTER DELAY

```

1251 .TITLE MISCELLANEOUS SECTIONS
1252 .SBTTL IDENTIFICATION
1253
1254
1255 .SBTTL REPORT CODING SECTION
1256
1257 005250 BGNRPT
1258 005250 L$RPT::
1259 005250 ENDRPT
1260 005250 L10011:
1261 005250 104425 TRAP C$RPT
1262
1263
1264 .SBTTL INITIALIZE SECTION
1265
1266 :++
1267 : THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
1268 : AT THE BEGINNING OF EACH PASS.
1269 :--
1270
1271 005252 BGNINIT
1272 005252 L$INIT::
1273 005252 GPWARD #0,R1 ;GET POINTER TO BASE ADDRESS OF P-TABLE
1274 005252 012700 000000 MOV #0,R0
1275 005256 104442 TRAP C$GPWARD
1276 005260 010001 MOV R0,R1
1277 005262 016137 000002 002274 MOV 2(R1),VECT ;GET INTERRUPT VECTOR
1278 005270 016137 000004 002304 MOV 4(R1),PRIOR ;GET PRIORITY LEVEL
1279 005276 016137 000006 002276 MOV 6(R1),SWSET ;GET ROCKER SWITCH SETTINGS
1280 005304 016137 000010 002240 MOV 10(R1),KDF11B ;GET KDF11-B INDICATOR
1281 005312 SETPRI #PRI07 ;INHIBIT INTERRUPTS
1282 005312 012700 000340 MOV #PRI07,R0
1283 005316 104441 TRAP C$SPRI
1284
1285
1286 005320 ENDINIT
1287 005320 L10012:
1288 005320 104411 TRAP C$INIT
1289
1290 005322 BGNPROT
1291 005322 L$PROT::
1292 005322 177777 .WORD -1 ;CSR OFFSET
1293 005324 177777 .WORD -1 ;MASS BUS OFFSET
1294 005326 177777 .WORD -1 ;DRIVE OFFSET
1295 005330 ENDPROT
1296
1297 005330 BGNAUTO
1298 005330 L$AUTO::
1299 005330 ENDAUTO
1300 005330 L10014:
1301 005330 104461 TRAP C$AUTO
  
```

1302
 1303
 1304
 1305
 1306
 1307
 1308
 1309
 1310
 1311
 1312
 1313
 1314
 1315
 1316
 1317
 1318
 1319
 1320
 1321
 1322
 1323
 1324
 1325
 1326
 1327
 1328
 1329
 1330
 1331
 1332
 1333

005332
 005332
 005332 005037 177520
 005336 005037 177522
 005342 012737 000001 002314
 005350 005037 002312
 005354 005037 015742
 005360 012737 000001 002254
 005366 005237 002264
 005372 005237 002266
 005376
 005376 013700 002274
 005402 104436
 005404
 005404 104432
 005406 000002
 005410
 005410
 005410 104412

```

.SBTTL CLEANUP CODING SECTION

:++
: THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED
: AT THE END OF EACH PASS.
:--

          BGNCLN
L$CLEAN::

          CLR      PCR          ;CLEAR PAGE CONTROL REGISTER
          CLR      RWREG       ;CLEAR READ/WRITE REGISTER
          MOV      #1,RSET     ;RESTORE DEFAULT VALUE
          CLR      RESPND      ;RESTORE DEFAULT
          CLR      ADDON       ;RESTORE DEFAULT
          MOV      #1,ANSR     ;RESTORE DEFAULT
          INC      PASS        ;INCREMENT PASS COUNT
          INC      PASCT       ;INCREMENT TEST 4 PASS COUNT
          CLRVEC  VECT        ;CLEAR INTERRUPT VECTOR
          MOV      VECT,R0
          TRAP    C$CVEC

          EXIT     CLN
          TRAP    C$EXIT
          .WORD   L10015-

          ENDCLN
L10015:   TRAP    C$CLEAN
  
```

```

1334      .TITLE HARDWARE TESTS
1335      .SBTTL IDENTIFICATION
1336
1337
1338      .SBTTL TEST 1: READ/WRITE REGISTER TEST
1339      :++
1340      :TEST TO VERIFY THAT THE READ/WRITE REGISTER AT ADDRESS 177522
1341      :IS WORD AND BYTE ADDRESSABLE.
1342      :--
1343
1344      177522      RWREG=177522
1345
1346      005412      BGNTST
1347
1348      005412      BGNSUB
1349      005412      104402      TRAP      C$BSUB
1350      005414      005037      177522      CLR      RWREG      ;LOAD ALL ZEROS
1351      005420      001412      BEQ      1$          ;BR IF CLEAR
1352      005422      005001      CLR      R1          ;EXPECTED DATA
1353      005424      013702      177522      MOV      RWREG,R2    ;COPY CONTENTS
1354      005430      ERRDF      1,RWR,RERR1 ;REGISTER CANNOT HOLD ALL ZEROS
1355      005430      104455      TRAP      C$ERDF
1356      005432      000001      .WORD    1
1357      005434      002570      .WORD    RWR
1358      005436      003712      .WORD    RERR1
1359      005440      CKLOOP      ;LOOP ON ERROR IF SELECTED
1360      005440      104406      TRAP      C$CLP1
1361      005442      EXIT      TST      ;ABORT TEST IF LOOP ON ERROR NOT SELECTED
1362      005442      104432      TRAP      C$EXIT
1363      005444      000610      .WORD    L10016-.
1364      005446      1$:      CKLOOP      ;LOOP ON ERROR IF SELECTED
1365      005446      104406      TRAP      C$CLP1
1366      005450      L10017:      ENDSUB
1367      005450      TRAP      C$ESUB
1368      005450      104403
1369
1370      005452      BGNSUB
1371      005452      104402      TRAP      C$BSUB
1372      005454      012737      177777      177522      MOV      #-1,RWREG ;LOAD ALL ONES
1373      005462      022737      177777      177522      CMP      #177777,RWREG ;CHECK THE REGISTER
1374      005470      001413      BEQ      2$          ;BR IF HOLDING GOOD DATA
1375      005472      012701      177777      MOV      #-1,R1      ;EXPECTED DATA
1376      005476      013702      177522      MOV      RWREG,R2    ;COPY CONTENTS
1377      005502      ERRDF      2,RWR,RERR2 ;REGISTER CANNOT HOLD ALL ONES
1378      005502      104455      TRAP      C$ERDF
1379      005504      000002      .WORD    2
1380      005506      002570      .WORD    RWR
1381      005510      003760      .WORD    RERR2
1382      005512      CKLOOP      ;LOOP ON ERROR IF SELECTED
1383      005512      104406      TRAP      C$CLP1
1384      005514      EXIT      TST      ;ABORT TEST IF ERROR AND NO LOOPING
1385      005514      104432      TRAP      C$EXIT
1386      005516      000536      .WORD    L10016-.
1387      005520      2$:      CKLOOP      ;LOOP ON ERROR IF SELECTED
1388      005520      104406      TRAP      C$CLP1
1389      005522      ENDSUB

```



```

1390 005522
1391 005522 104403
1392
1393 005524
1394 005524 104402
1395 005526 012737 125252 177522
1396 005534 022737 125252 177522
1397 005542 001413
1398 005544 012701 125252
1399 005550 013702 177522
1400 005554
1401 005554 104455
1402 005556 000003
1403 005560 002570
1404 005562 004026
1405 005564
1406 005564 104406
1407 005566
1408 005566 104432
1409 005570 000464
1410 005572
1411 005572 104406
1412 005574
1413 005574
1414 005574 104403
1415
1416 005576
1417 005576 104402
1418 005600 105037 177522
1419 005604 022737 125000 177522
1420 005612 001413
1421 005614 012701 125000
1422 005620 013702 177522
1423 005624
1424 005624 104455
1425 005626 000004
1426 005630 002570
1427 005632 004074
1428 005634
1429 005634 104406
1430 005636
1431 005636 104432
1432 005640 000414
1433 005642
1434 005642 104406
1435 005644
1436 005644
1437 005644 104403
1438
1439 005646
1440 005646 104402
1441 005650 000337 177522
1442 005654 022737 000252 177522
1443 005662 001407
1444 005664
1445 005664 104455

L10020:
TRAP C$ESUB
BGNSUB
TRAP C$BSUB
MOV #125252,RWREG ;LOAD ALTERNATING 1'S AND 0'S BIT PATTERN
CMP #125252,RWREG ;CHECK DATA
BEQ 3$ ;BR IF GOOD
MOV #125252,R1 ;EXPECTED DATA
MOV RWREG,R2 ;COPY CONTENTS
ERRDF 3,RWR,RERR3 ;CANNOT HOLD GOOD DATA
TRAP C$ERDF
.WORD 3
.WORD RWR
.WORD RERR3
CKLOOP ;LOOP ON ERROR IF SELECTED
TRAP C$CLP1
EXIT TST ;ABORT TEST IF ERROR DETECTED
TRAP C$EXIT
.WORD L10016-.
3$: CKLOOP ;CHECK FOR LOOP ON ERROR AGAIN
TRAP C$CLP1
ENDSUB

L10021:
TRAP C$ESUB
BGNSUB
TRAP C$BSUB
CLRB RWREG ;CLEAR THE REGISTER'S LOW BYTE
CMP #125000,RWREG ;DID IT CLEAR PROPERLY?
BEQ 4$ ;BR IF YES
MOV #125000,R1 ;EXPECTED DATA
MOV RWREG,R2 ;COPY CONTENTS
ERRDF 4,RWR,RERR4 ;DID NOT RESPOND PROPERLY TO BYTE INSTRUCTION
TRAP C$ERDF
.WORD 4
.WORD RWR
.WORD RERR4
CKLOOP ;LOOP ON ERROR IF SELECTED
TRAP C$CLP1
EXIT TST ;ABORT TEST IF ERROR DETECTED
TRAP C$EXIT
.WORD L10016-.
4$: CKLOOP ;CHECK FOR LOOP ON ERROR AGAIN
TRAP C$CLP1
ENDSUB

L10022:
TRAP C$ESUB
BGNSUB
TRAP C$BSUB
SWAB RWREG ;SWAP BYTES IN THE REGISTER
CMP #252,RWREG ;GOOD DATA?
BEQ 5$ ;BR IF YES
ERRDF 5,RWR,RERR4 ;BYTE INSTRUCTION ERROR
TRAP C$ERDF

```

1446	005666	000005			.WORD	5		
1447	005670	002570			.WORD	RWR		
1448	005672	004074			.WORD	RERR4		
1449	005674				CKLOOP			:LOOP ON ERROR IF SELECTED
1450	005674	104406			TRAP	C\$CLP1		
1451	005676				EXIT	TST		:ABORT TEST IF ERROR DETECTED
1452	005676	104432			TRAP	C\$EXIT		
1453	005700	000354			.WORD	L10016-		
1454	005702		5\$:		CKLOOP			:CHECK FOR LOOP ON ERROR AGAIN
1455	005702	104406			TRAP	C\$CLP1		
1456	005704				ENDSUB			
1457	005704			L10023:				
1458	005704	104403			TRAP	C\$ESUB		
1459								
1460	005706				BGNSUB			
1461	005706	104402			TRAP	C\$BSUB		
1462	005710	012737	052525	177522	MOV	#052525,RWREG		:LOAD AN ALTERNATING 0'S AND 1'S BIT PATTERN
1463	005716	022737	052525	177522	CMP	#052525,RWREG		:CHECK IT
1464	005724	001413			BEQ	6\$:BR IF GOOD DATA
1465	005726	012701	052525		MOV	#052525,R1		:EXPECTED DATA
1466	005732	013702	177522		MOV	RWREG,R2		:COPY CONTENTS
1467	005736				ERRDF	6,RWR,RERR3		:CANNOT HOLD GOOD DATA
1468	005736	104455			TRAP	C\$ERDF		
1469	005740	000006			.WORD	6		
1470	005742	002570			.WORD	RWR		
1471	005744	004026			.WORD	RERR3		
1472	005746				CKLOOP			:LOOP ON ERROR IF SELECTED
1473	005746	104406			TRAP	C\$CLP1		
1474	005750				EXIT	TST		:ABORT TEST IF ERROR DETECTED
1475	005750	104432			TRAP	C\$EXIT		
1476	005752	000302			.WORD	L10016-		
1477	005754		6\$:		CKLOOP			:CHECK FOR LOOP ON ERROR AGAIN
1478	005754	104406			TRAP	C\$CLP1		
1479	005756				ENDSUB			
1480	005756			L10024:				
1481	005756	104403			TRAP	C\$ESUB		
1482								
1483	005760				BGNSUB			
1484	005760	104402			TRAP	C\$BSUB		
1485	005762	105037	177523		CLRB	RWREG+1		:CLEAR HIGH BYTE OF REGISTER
1486	005766	022737	000125	177522	CMP	#125,RWREG		:CHECK THE RESULTING CONTENTS OF THE REGISTER
1487	005774	001413			BEQ	7\$:BR IF GOOD DATA
1488	005776	012701	000125		MOV	#125,R1		:EXPECTED DATA
1489	006002	013702	177522		MOV	RWREG,R2		:COPY CONTENTS
1490	006006				ERRDF	7,RWR,RERR4		:BYTE INSTRUCTION ERROR
1491	006006	104455			TRAP	C\$ERDF		
1492	006010	000007			.WORD	7		
1493	006012	002570			.WORD	RWR		
1494	006014	004074			.WORD	RERR4		
1495	006016				CKLOOP			:LOOP ON ERROR IF SELECTED
1496	006016	104406			TRAP	C\$CLP1		
1497	006020				EXIT	TST		:ABORT TEST IF ERROR DETECTED
1498	006020	104432			TRAP	C\$EXIT		
1499	006022	000232			.WORD	L10016-		
1500	006024		7\$:		CKLOOP			:CHECK FOR LOOP ON ERROR AGAIN
1501	006024	104406			TRAP	C\$CLP1		

```

1502 006026          ENDSUB
1503 006026          L10025: TRAP    C$ESUB
1504 006026 104403
1505
1506 006030          BGNSUB
1507 006030 104402   TRAP    C$BSUB
1508 006032 000337 177522   SWAB   RWREG      ;SWAP BYTES
1509 006036 022737 052400 177522   CMP    #052400,RWREG ;DATA GOOD?
1510 006044 001413          BEQ    10$         ;BR IF YES
1511 006046 012701 052400   MOV    #52400,R1   ;EXPECTED DATA
1512 006052 013702 177522   MOV    RWREG,R2   ;COPY CONTENTS
1513 006056          ERRDF  10,RWR,RERR4 ;BYTE INSTRUCTION ERROR
1514 006056 104455   TRAP    C$ERDF
1515 006060 000012   .WORD  10
1516 006062 002570   .WORD  RWR
1517 006064 004074   .WORD  RERR4
1518 006066          CKLOOP
1519 006066 104406   TRAP    C$CLP1    ;LOOP ON ERROR IF SELECTED
1520 006070          EXIT    TST        ;ABORT TEST IF ERROR DETECTED
1521 006070 104432   TRAP    C$EXIT
1522 006072 000162   .WORD  L10016-.
1523 006074          10$:  CKLOOP
1524 006074 104406   TRAP    C$CLP1    ;CHECK FOR LOOP ON ERROR AGAIN
1525 006076          ENDSUB
1526 006076          I10026:
1527 006076 104403   TRAP    C$ESUB
1528
1529 006100          BGNSUB
1530 006100 104402   TRAP    C$BSUB
1531 006102 005037 177522   CLR    RWREG      ;MAKE SURE THE C-BIT IS CLEAR
1532 006106 052737 100000 177522   BIS    #BIT15,RWREG ;SET MSB
1533 006114 013703 177522   MOV    RWREG,R3   ;COPY DATA IN RWREG
1534 006120 023703 177522   ROTLP1: CMP    RWREG,R3  ;ARE THEY THE SAME?
1535 006124 001005          BNE    11$         ;BR IF NO
1536 006126 006003          ROR    R3         ;ROTATE THE SET BIT
1537 006130 001412          BEQ    12$         ;BR WHEN FINISHED
1538 006132 006037 177522   ROR    RWREG      ;REPEAT ROTATE
1539 006136 000770          BR     ROTLP1     ;LOOP UNTIL ROTATE IS COMPLETE
1540 006140          11$:  ERRDF  11,RWR,RERR5
1541 006140 104455   TRAP    C$ERDF
1542 006142 000013   .WORD  11
1543 006144 002570   .WORD  RWR
1544 006146 004142   .WORD  RERR5
1545 006150          CKLOOP
1546 006150 104406   TRAP    C$CLP1    ;LOOP ON ERROR IF SELECTED
1547 006152          EXIT    TST        ;SKIP REST OF TEST
1548 006152 104432   TRAP    C$EXIT
1549 006154 000100   .WORD  L10016-.
1550 006156          12$:  CKLOOP
1551 006156 104406   TRAP    C$CLP1    ;CHECK FOR LOOP ON ERROR
1552 006160          ENDSUB
1553 006160          L10027:
1554 006160 104403   TRAP    C$ESUB
1555
1556 006162          BGNSUB
1557 006162 104402   TRAP    C$BSUB

```

1558	006164	012737	177777	177522		MOV	#-1,RWREG	:SET ALL ONES
1559	006172	042737	100000	177522		BIC	#BIT15,RWREG	:CLEAR MSB
1560	006200	013703	177522			MOV	RWREG,R3	:COPY DATA
1561	006204	023703	177522		ROTLP2:	CMP	RWREG,R3	:ARE THEY THE SAME?
1562	006210	001010				BNE	13\$:BR IF NO
1563	006212	000261				SEC		:SET C-BIT FOR ROTATE
1564	006214	006037	177522			ROR	RWREG	:ROTATE CLEAR BIT
1565	006220	006003				ROR	R3	:REPEAT
1566	006222	022703	077777			CMP	#077777,R3	:FINISHED?
1567	006226	001366				BNE	ROTLP2	:BR IF NOT YET
1568	006230	000407				BR	14\$:SUBTEST FINISHED
1569	006232				13\$:	ERRDF	12,RWR,RERR6	
1570	006232	104455				TRAP	C\$ERDF	
1571	006234	000014				.WORD	12	
1572	006236	002570				.WORD	RWR	
1573	006240	004164				.WORD	RERR6	
1574	006242					CKLOOP		:LOOP ON ERROR IF SELECTED
1575	006242	104406				TRAP	C\$CLP1	
1576	006244					EXIT TST		
1577	006244	104432				TRAP	C\$EXIT	
1578	006246	000006				.WORD	L10016-	
1579	006250				14\$:	CKLOOP		
1580	006250	104406				TRAP	C\$CLP1	
1581	006252					ENDSUB		
1582	006252				I 10030:			
1583	006252	104403				TRAP	C\$ESUB	
1584								
1585	006254					ENDTST		
1586	006254				L10016:			
1587	006254	104401				TRAP	C\$ETST	

```

1588 .SBTTL TEST 2: PAGE CONTROL REGISTER TEST
1589 :++
1590 :TEST TO VERIFY THAT THE PAGE CONTROL REGISTER IS WORD
1591 :AND BYTE ADDRESSABLE.
1592 :--
1593
1594 006256 BGNTST
1595
1596 006256 005737 002240 TST KDF11B ;IF THIS IS A KDF11-B...
1597 006262 001402 BEQ 15$ ;...THEN SKIP THIS TEST
1598 006264 EXIT TST
1599 006264 104432 TRAP C$EXIT
1600 006266 000654 .WORD L10031-.
1601
1602 006270 15$: BGNSUB
1603 006270 104402 TRAP C$BSUB
1604 006272 005037 177520 CLR PCR ;LOAD ALL ZEROS
1605 006276 001412 BEQ 1$ ;BR IF CLEARED
1606 006300 005001 CLR R1 ;EXPECTED DATA
1607 006302 013702 177520 MOV PCR,R2 ;COPY CONTENTS
1608 006306 ERRDF 13,PACR,RERR1 ;REGISTER CANNOT HOLD ALL ZEROS
1609 006306 104455 TRAP C$ERDF
1610 006310 000015 .WORD 13
1611 006312 002634 .WORD PACR
1612 006314 003712 .WORD RERR1
1613 006316 CKLOOP ;LOOP ON ERROR IF SELECTED
1614 006316 104406 TRAP C$CLP1
1615 006320 EXIT TST ;ABORT TEST IF ERROR DETECTED
1616 006320 104432 TRAP C$EXIT
1617 006322 000620 .WORD L10031-.
1618 006324 1$: CKLOOP ;CHECK FOR LOOP ON ERROR AGAIN
1619 006324 104406 TRAP C$CLP1
1620 006326 ENDSUB
1621 006326 L10032:
1622 006326 104403 TRAP C$ESUB
1623
1624 006330 BGNSUB
1625 006330 104402 TRAP C$BSUB
1626 006332 012737 177777 177520 MOV #-1,PCR ;LOAD ALL ONES
1627 006340 022737 177777 177520 CMP #177777,PCR ;CHECK FOR GOOD DATA
1628 006346 001413 BEQ 2$ ;BR IF GOOD
1629 006350 012701 177777 MOV #-1,R1 ;EXPECTED DATA
1630 006354 013702 177520 MOV PCR,R2 ;COPY CONTENTS
1631 006360 ERRDF 14,PACR,RERR2 ;REGISTER CANNOT HOLD ALL ONES
1632 006360 104455 TRAP C$ERDF
1633 006362 000016 .WORD 14
1634 006364 002634 .WORD PACR
1635 006366 003760 .WORD RERR2
1636 006370 CKLOOP ;LOOP ON ERROR IF SELECTED
1637 006370 104406 TRAP C$CLP1
1638 006372 EXIT TST ;ABORT TEST IF ERROR DETECTED
1639 006372 104432 TRAP C$EXIT
1640 006374 000546 .WORD L10031-.
1641 006376 2$: CKLOOP ;CHECK FOR LOOP ON ERROR AGAIN
1642 006376 104406 TRAP C$CLP1
1643 006400 ENDSUB

```

```

1644 006400          L10033:
1645 006400 104403   TRAP    C$ESUB
1646
1647
1648 006402          BGNSUB
1649 006402 104402   TRAP    C$BSUB
1650 006404 012737 125252 177520  MOV    #125252,PCR    ;LOAD AN ALTERNATING 1'S AND 0'S BIT PATTERN
1651 006412 022737 125252 177520  CMP    #125252,PCR    ;CHECK THE RESULTS
1652 006420 001413          BEQ    3$             ;BR IF GOOD DATA
1653 006422 012701 125252          MOV    #125252,R1    ;EXPECTED DATA
1654 006426 013702 177520          MOV    PCR,R2        ;COPY CONTENTS
1655 006432          ERRDF  15,PACR,RERR3 ;REGISTER CANNOT HOLD GOOD DATA
1656 006432 104455   TRAP    C$ERDF
1657 006434 000017          .WORD  15
1658 006436 002634          .WORD  PACR
1659 006440 004026          .WORD  RERR3
1660 006442          CKLOOP
1661 006442 104406   TRAP    C$CLP1       ;LOOP ON ERROR IF SELECTED
1662 006444          EXIT    TST          ;ABORT TEST IF ERROR DETECTED
1663 006444 104432   TRAP    C$EXIT
1664 006446 000474          .WORD  L10031-.
1665 006450          3$: CKLOOP
1666 006450 104406   TRAP    C$CLP1       ;CHECK FOR LOOP ON ERROR AGAIN
1667 006452          ENDSUB
1668 006452          I.10034:
1669 006452 104403   TRAP    C$ESUB
1670
1671 006454          BGNSUB
1672 006454 104402   TRAP    C$BSUB
1673 006456 105037 177520  CLRB   PCR           ;CLEAR THE REGISTER'S LOW BYTE
1674 006462 022737 125000 177520  CMP    #125000,PCR    ;COMPARE THE RESULTS
1675 006470 001413          BEQ    4$             ;BR IF GOOD DATA
1676 006472 012701 125000          MOV    #125000,R1    ;EXPECTED DATA
1677 006476 013702 177520          MOV    PCR,R2        ;COPY CONTENTS
1678 006502          ERRDF  16,PACR,RERR4 ;BYTE INSTRUCTION ERROR
1679 006502 104455   TRAP    C$ERDF
1680 006504 000020          .WORD  16
1681 006506 002634          .WORD  PACR
1682 006510 004074          .WORD  RERR4
1683 006512          CKLOOP
1684 006512 104406   TRAP    C$CLP1       ;LOOP ON ERROR IF SELECTED
1685 006514          EXIT    TST          ;ABORT TEST IF ERROR DETECTED
1686 006514 104432   TRAP    C$EXIT
1687 006516 000424          .WORD  L10031-.
1688 006520          4$: CKLOOP
1689 006520 104406   TRAP    C$CLP1       ;CHECK FOR LOOP ON ERROR
1690 006522          ENDSUB
1691 006522          L10035:
1692 006522 104403   TRAP    C$ESUB
1693
1694 006524          BGNSUB
1695 006524 104402   TRAP    C$BSUB
1696 006526 000337 177520  SWAB   PCR           ;SWAP BYTES
1697 006532 022737 000252 177520  CMP    #252,PCR      ;CHECK THE RESULTS
1698 006540 001413          BEQ    5$             ;BR IF GOOD DATA
1699 006542 012701 000252          MOV    #252,R1      ;EXPECTED DATA

```

1700	006546	013702	177520	MOV	PCR,R2	:COPY CONTENTS
1701	006552			ERRDF	17,PACR,RERR4	:BYTE INSTRUCTION ERROR
1702	006552	104455		TRAP	C\$ERDF	
1703	006554	000021		.WORD	17	
1704	006556	002634		.WORD	PACR	
1705	006560	004074		.WORD	RERR4	
1706	006562			CKLOOP		:LOOP ON ERROR IF SELECTED
1707	006562	104406		TRAP	C\$CLP1	
1708	006564			EXIT	TST	:ABORT TEST IF ERROR DETECTED
1709	006564	104432		TRAP	C\$EXIT	
1710	006566	000354		.WORD	L10031-.	
1711	006570			5\$: CKLOOP		:CHECK FOR LOOP ON ERROR
1712	006570	104406		TRAP	C\$CLP1	
1713	006572			ENDSUB		
1714	006572			L10036:		
1715	006572	104403		TRAP	C\$ESUB	
1716	006574			BGNSUB		
1717	006574	104402		TRAP	C\$BSUB	
1718	006576	012737	052525 177520	MOV	#052525,PCR	:LOAD AN ALTERNATING 0'S AND 1'S BIT PATTERN
1719	006604	022737	052525 177520	CMP	#052525,PCR	:CHECK THE RESULTS
1720	006612	001413		BEQ	6\$:BR IF GOOD DATA
1721	006614	012701	052525	MOV	#052525,R1	:EXPECTED DATA
1722	006620	013702	177520	MOV	PCR,R2	:COPY CONTENTS
1723	006624			ERRDF	20,PACR,RERR3	:REGISTER CANNOT HOLD GOOD DATA
1724	006624	104455		TRAP	C\$ERDF	
1725	006626	000024		.WORD	20	
1726	006630	002634		.WORD	PACR	
1727	006632	004026		.WORD	RERR3	
1728	006634			CKLOOP		:LOOP ON ERROR IF SELECTED
1729	006634	104406		TRAP	C\$CLP1	
1730	006636			EXIT	TST	:ABORT TEST IF ERROR DETECTED
1731	006636	104432		TRAP	C\$EXIT	
1732	006640	000302		.WORD	L10031-.	
1733	006642			6\$: CKLOOP		:CHECK FOR LOOP ON ERROR
1734	006642	104406		TRAP	C\$CLP1	
1735	006644			ENDSUB		
1736	006644			L10037:		
1737	006644	104403		TRAP	C\$ESUB	
1738	006646			BGNSUB		
1739	006646	104402		TRAP	C\$BSUB	
1740	006650	105037	177521	CLRB	PCR+1	:CLEAR THE HIGH BYTE
1741	006654	022737	000125 177520	CMP	#125,PCR	:CHECK THE REGISTER CONTENTS
1742	006662	001413		BEQ	7\$:BR IF GOOD DATA
1743	006664	012701	000125	MOV	#125,R1	:EXPECTED DATA
1744	006670	013702	177520	MOV	PCR,R2	:COPY CONTENTS
1745	006674			ERRDF	21,PACR,RERR4	:BYTE INSTRUCTION ERROR
1746	006674	104455		TRAP	C\$ERDF	
1747	006676	000025		.WORD	21	
1748	006700	002634		.WORD	PACR	
1749	006702	004074		.WORD	RERR4	
1750	006704			CKLOOP		:LOOP ON ERROR IF SELECTED
1751	006704	104406		TRAP	C\$CLP1	
1752	006706			EXIT	TST	:ABORT TEST IF ERROR DETECTED
1753	006706	104432		TRAP	C\$EXIT	
1754	006710	000232		.WORD	L10031-.	
1755	006712			7\$: CKLOOP		:CHECK FOR LOOP ON ERROR

1756	006712	104406			TRAP	C\$CLP1	
1757	006714				ENDSUB		
1758	006714				L10040:		
1759	006714	104403			TRAP	C\$ESUB	
1760							
1761	006716				BGNSUB		
1762	006716	104402			TRAP	C\$BSUB	
1763	006720	000337	177520		SWAB	PCR	:SWAP BYTES
1764	006724	022737	052400	177520	CMP	#052400,PCR	:CHECK RESULTING CONTENTS
1765	006732	001413			BEQ	10\$:BR IF GOOD DATA
1766	006734	012701	052400		MOV	#52400,R1	:EXPECTED DATA
1767	006740	013702	177520		MOV	PCR,R2	:COPY CONTENTS
1768	006744				ERRDF	22,PACR,RERR4	:BYTE INSTRUCTION ERROR
1769	006744	104455			TRAP	C\$ERDF	
1770	006746	000026			.WORD	22	
1771	006750	002634			.WORD	PACR	
1772	006752	004074			.WORD	RERR4	
1773	006754				CKLOOP		:LOOP ON ERROR IF SELECTED
1774	006754	104406			TRAP	C\$CLP1	
1775	006756				EXIT	TST	:ABORT TEST IF ERROR DETECTED
1776	006756	104432			TRAP	C\$EXIT	
1777	006760	000162			.WORD	L10031-	
1778	006762				10\$:	CKLOOP	:CHECK FOR LOOP ON ERROR
1779	006762	104406			TRAP	C\$CLP1	
1780	006764				ENDSUB		
1781	006764				L10041:		
1782	006764	104403			TRAP	C\$ESUB	
1783							
1784	006766				BGNSUB		
1785	006766	104402			TRAP	C\$BSUB	
1786	006770	005037	177520		CLR	PCR	:MAKE SURE THE C-BIT IS CLEAR
1787	006774	052737	100000	177520	BIS	#BIT15,PCR	:SET MSB
1788	007002	013703	177520		MOV	PCR,R3	:COPY DATA IN PCR
1789	007006	023703	177520		ROTLP3:	CMP	PCR,R3
1790	007012	001005			BNE	11\$:ARE THEY THE SAME?
1791	007014	006003			ROR	R3	:BR IF NO
1792	007016	001412			BEQ	12\$:ROTATE THE SET BIT
1793	007020	006037	177520		ROR	PCR	:BR IF FINISHED
1794	007024	000770			BR	ROTLP3	:REPEAT ROTATE
1795	007026				11\$:	ERRDF	:LOOP UNTIL ROTATE IS COMPLETE
1796	007026	104455			TRAP	23,PACR,RERR5	
1797	007030	000027			TRAP	C\$ERDF	
1798	007032	002634			.WORD	23	
1799	007034	004142			.WORD	PACR	
1800	007036				.WORD	RERR5	
1801	007036	104406			CKLOOP		:LOOP ON ERROR IF SELECTED
1802	007040				TRAP	C\$CLP1	
1803	007040	104432			EXIT	TST	:SKIP REST OF TEST
1804	007042	000100			TRAP	C\$EXIT	
1805	007044				.WORD	L10031-	
1806	007044	104406			12\$:	CKLOOP	:CHECK FOR LOOP ON ERROR
1807	007046				TRAP	C\$CLP1	
1808	007046				ENDSUB		
1809	007046	104403			L10042:		
1810					TRAP	C\$ESUB	
1811	007050				BGNSUB		

1812	007050	104402			TRAP	C\$BSUB	
1813	007052	012737	177777	177520	MOV	#-1,PCR	:SET ALL ONES
1814	007060	042737	100000	177520	BIC	#BIT15,PCR	:CLEAR MSB
1815	007066	013703	177520		MOV	PCR,R3	:COPY DATA
1816	007072	023703	177520		ROTLP4: CMP	PCR,R3	:ARE THEY THE SAME?
1817	007076	001010			BNE	13\$:BR IF NO
1818	007100	000261			SEC		:SET C-BIT FOR ROTATE
1819	007102	006037	177520		ROR	PCR	:ROTATE CLEAR BIT
1820	007106	006003			ROR	R3	:REPEAT
1821	007110	022703	077777		CMP	#077777,R3	:ALL ONES?
1822	007114	001366			BNE	ROTLP4	:BR IF NOT YET
1823	007116	000407			BR	14\$:SUBTEST FINISHED
1824	007120				13\$: ERRDF	24,PACR,RERR6	
1825	007120	104455			TRAP	C\$ERDF	
1826	007122	000030			.WORD	24	
1827	007124	002634			.WORD	PACR	
1828	007126	004164			.WORD	RERR6	
1829	007130				CKLOOP		:LOOP ON ERROR IF SELECTED
1830	007130	104406			TRAP	C\$CLP1	
1831	007132				EXIT TST		
1832	007132	104432			TRAP	C\$EXIT	
1833	007134	000006			.WORD	L10031-	
1834	007136				14\$: CKLOOP		
1835	007136	104406			TRAP	C\$CLP1	
1836	007140				ENDSUB		
1837	007140				L10043: TRAP	C\$ESUB	
1838	007140	104403			ENDTST		
1839	007142				L10031: TRAP	C\$ETST	
1840	007142						
1841	007142	104401					
1842							

```

1843 .SBTTL TEST 3: BEVENT CLAMP ENABLE TEST
1844 :++
1845 :TEST TO VERIFY THAT THE BEVENT CLAMP CAN BE ENABLED. (IF TESTING A BDV11, THIS
1846 :TEST ASSUMES THAT SWITCH #5 OF E21 IS IN THE ON POSITION, AND THE M8012
1847 :MODULE IS LOCATED IN THE SAME BACKPLANE THAT THE LINE TIME CLOCK
1848 :IS GENERATED FROM.) CHECKS PRIORITY INTERRUPT LEVEL 6 IF IT WAS
1849 :CHANGED IN HARDWARE TABLE AND IF THE DEVICE UNDER TEST IS KDF11-B.
1850
1851 :--
1852
1853 007144 BGNTST
1854
1855 177546 BEVREG=177546
1856
1857 007144 005737 002264 TST PASS ;IF THIS IS FIRST PASS
1858 007150 001402 BEQ 1$ ;THEN DO THE TEST
1859 007152 EXIT TST ;ELSE DON'T
1860 007152 104432 TRAP C$EXIT
1861 007154 001362 .WORD L10044-.
1862 007156 005037 007650 1$: CLR ICOUNT
1863 007162 BGNSUB
1864 007162 104402 TRAP C$SUB
1865 007164 SETVEC VECT,#INTSR,#PRI07 ;SET INTERRUPT VECTOR,INHIBIT INTERRUPTS
1866 007164 012746 000340 MOV #PRI07,-(SP)
1867 007170 012746 007642 MOV #INTSR,-(SP)
1868 007174 013746 002274 MOV VECT,-(SP)
1869 007200 012746 000003 MOV #3,-(SP)
1870 007204 104437 TRAP C$SVEC
1871 007206 062706 000010 ADD #10,SP
1872 007212 052737 000100 177546 BIS #BIT06,BEVREG ;REMOVE BEVENT CLAMP
1873 007220 SETPRI #PRI00 ;ALLOW INTERRUPTS
1874 007220 012700 000000 MOV #PRI00,R0
1875 007224 104441 TRAP C$SPRI
1876 007226 004537 005212 JSR R5,WDELAY ;DELAY APPROX. 40 MSECS.
1877 007232 000050 40.
1878 007234 SETPRI #PRI07 ;INHIBIT FURTHER INTERRUPTS
1879 007234 012700 000340 MOV #PRI07,R0
1880 007240 104441 TRAP C$SPRI
1881 007242 022737 000002 007650 CMP #2,ICOUNT ;DID THE MINIMUM OF TWO INTERRUPTS OCCUR?
1882 007250 003404 BLE 2$ ;BR IF YES
1883 007252 ERRDF 25,BVERR1 ;BEVENT CLAMP ENABLE FAILED
1884 007252 104455 TRAP C$ERDF
1885 007254 000031 .WORD 25
1886 007256 000000 .WORD 0
1887 007260 007652 .WORD BVERR1
1888 007262 2$: CKLOOP ;CHECK FOR LOOP ON ERROR
1889 007262 104406 TRAP C$CLP1
1890 007264 005037 007650 CLR ICOUNT ;CLEAR INTERRUPT COUNT
1891 007270 ENDSUB
1892 007270 L10045:
1893 007270 104403 TRAP C$ESUB
1894
1895 007272 BGNSUB
1896 007272 104402 TRAP C$SUB
1897 007274 042737 000100 177546 BIC #BIT06,BEVREG ;SET BEVENT CLAMP
1898 007302 SETPRI #PRI00 ;ALLOW INTERRUPTS
  
```

HARDWARE TESTS MACY11 30(1046)
CVMBAD.P11 10-JUL-81 17:08

20-JUL-81 11:44 PAGE 42
TEST 3: BEVENT CLAMP ENABLE TEST

SEQ 0042

1899	007302	012700	000000		MOV	#PRI00,RO	
1900	007306	104441			TRAP	C\$SPRI	
1901	007310	004537	005212		JSR	R5,WDELAY	:DELAY APPROX. 40 MSECS
1902	007314	000050			40.		:DELAY COUNT
1903	007316				SETPRI	#PRI07	:SET HIGHEST PRIORITY
1904	007316	012700	000340		MOV	#PRI07,RO	
1905	007322	104441			TRAP	C\$SPRI	
1906	007324	022737	000001	007650	CMP	#1,ICOUNT	:CHECK INTERRUPT COUNT
1907	007332	002004			BGE	4\$:BR IF NO INTERRUPTS OCCURRED
1908	007334				ERRDF	26,,BVERR2	:BEVENT CLAMP DID NOT PREVENT INTERRUPTS
1909	007334	104455			TRAP	C\$ERDF	
1910	007336	000032			.WORD	26	
1911	007340	000000			.WORD	0	
1912	007342	007720			.WORD	BVERR2	
1913	007344			4\$:	CKLOOP		:CHECK FOR LOOP ON ERROR
1914	007344	104406			TRAP	C\$CLP1	
1915	007346	005037	007650		CLR	ICOUNT	:CLEAR INTERRUPT COUNT
1916	007352				ENDSUB		
1917	007352			L10046:			
1918	007352	104403			TRAP	C\$ESUB	
1919							
1920	007354	022737	000006	002304	CMP	#6,PRIOR	:WAS PRIORITY CHANGED?
1921	007362	001405			BEQ	100\$:IF YES, BRANCH
1922	007364	005737	002240		TST	KDF11B	:KDF11B ?
1923	007370	001002			BNE	100\$:IF YES, BRANCH
1924	007372				EXIT	TST	:NO,EXIT
1925	007372	104432			TRAP	C\$EXIT	
1926	007374	001142			.WORD	L10044-	
1927							
1928	007376			100\$:	BGNSUB		
1929	007376	104402			TRAP	C\$BSUB	
1930	007400	052737	000100	177546	BIS	#BIT06,BEVREG	:REMOVE BEVENT CLAMP
1931	007406				SETPRI	#PRI05	:ALLOW INTERRUPTS AT 5
1932	007406	012700	000240		MOV	#PRI05,RO	
1933	007412	104441			TRAP	C\$SPRI	
1934	007414	004537	005212		JSR	R5,WDELAY	:DELAY APPROX. 40 MSECS.
1935	007420	000050			40.		:INHIBIT FURTHER INTERRUPTS
1936	007422				SETPRI	#PRI07	
1937	007422	012700	000340		MOV	#PRI07,RO	
1938	007426	104441			TRAP	C\$SPRI	
1939	007430	022737	000002	007650	CMP	#2,ICOUNT	:DID THE MINIMUM OF TWO INTERRUPTS OCCUR?
1940	007436	003404			BLE	101\$:BR IF YES
1941	007440				ERRDF	52,,BVERR4	:PRIORITY IS WRONG
1942	007440	104455			TRAP	C\$ERDF	
1943	007442	000064			.WORD	52	
1944	007444	000000			.WORD	0	
1945	007446	010034			.WORD	BVERR4	
1946	007450			101\$:	CKLOOP		:CHECK FOR LOOP ON ERROR
1947	007450	104406			TRAP	C\$CLP1	
1948	007452	005037	007650		CLR	ICOUNT	:CLEAR INTERRUPT COUNT
1949	007456				ENDSUB		
1950	007456			L10047:			
1951	007456	104403			TRAP	C\$ESUB	
1952							
1953	007460				BGNSUB		
1954	007460	104402			TRAP	C\$BSUB	

HARDWARE TESTS MACY11 30(1046) 20-JUL-81 11:44 PAGE 43
 CVMBAD.P11 10-JUL-81 17:08 TEST 3: BEVENT CLAMP ENABLE TEST

SEQ 0043

1955	007462	052737	000100	177546	BIS	#BIT06, BEVREG	:REMOVE BEVENT CLAMP
1956	007470				SETPRI	#PRI06	:DON'T ALLOW INTERRUPTS
1957	007470	012700	000300		MOV	#PRI06, R0	
1958	007474	104441			TRAP	C\$SPRI	
1959	007476	004537	005212		JSR	R5, WDELAY	:DELAY APPROX. 40 MSECS
1960	007502	000050			40.		:DELAY COUNT
1961	007504				SETPRI	#PRI07	:SET PRIORITY 7
1962	007504	012700	000340		MOV	#PRI07, R0	
1963	007510	104441			TRAP	C\$SPRI	
1964	007512	022737	000001	007650	CMP	#1, ICOUNT	:CHECK INTERRUPT COUNT
1965	007520	002004			BGE	102\$:BR IF NO INTERRUPTS OCCURRED
1966	007522				ERRDF	53, BVERR5	:PRIOR. 6 DIDN'T PREVENT INTERR.
1967	007522	104455			TRAP	C\$ERDF	
1968	007524	000065			.WORD	53	
1969	007526	000000			.WORD	0	
1970	007530	010102			.WORD	BVERR5	
1971	007532			102\$:	CKLOOP		:CHECK FOR LOOP ON ERROR
1972	007532	104406			TRAP	C\$CLP1	
1973	007534	005037	007650		CLR	ICOUNT	:CLEAR INTERRUPT COUNT
1974	007540				ENDSUB		
1975	007540			L10050:			
1976	007540	104403			TRAP	C\$ESUB	
1977							
1978	007542	005737	002240		TST	KDF11B	:KDF11B ?
1979	007546	001002			BNE	5\$	
1980	007550				EXIT	TST	
1981	007550	104432			TRAP	C\$EXIT	
1982	007552	000764			.WORD	L10044-	
1983							
1984	007554			5\$:	BGNSUB		:SET INTERRUPTS
1985	007554	104402			TRAP	C\$BSUB	
1986	007556	052737	000100	177546	BIS	#BIT06, BEVREG	
1987	007564				BRESET		:RESET INTERRUPTS
1988	007564	104433			TRAP	C\$RESET	
1989	007566				SETPRI	#PRI00	:WITH LOW PRIORITY
1990	007566	012700	000000		MOV	#PRI00, R0	
1991	007572	104441			TRAP	C\$SPRI	
1992	007574	004537	005212		JSR	R5, WDELAY	:DELAY APPROX. 40 MSECS
1993	007600	000050			40.		
1994	007602				SETPRI	#PRI07	
1995	007602	012700	000340		MOV	#PRI07, R0	
1996	007606	104441			TRAP	C\$SPRI	
1997	007610	005737	007650		TST	ICOUNT	:0 INTERRUPTS ?
1998	007614	001404			BEQ	6\$:IF YES BRANCH
1999	007616				ERRDF	51, BVERR3	:RESET DIDN'T WORK
2000	007616	104455			TRAP	C\$ERDF	
2001	007620	000063			.WORD	51	
2002	007622	000000			.WORD	0	
2003	007624	007766			.WORD	BVERR3	
2004	007626			6\$:	CKLOOP		:CHECK FOR LOOP ON ERROR
2005	007626	104406			TRAP	C\$CLP1	
2006	007630	005037	007650		CLR	ICOUNT	
2007	007634				ENDSUB		
2008	007634			L10051:			
2009	007634	104403			TRAP	C\$ESUB	
2010	007636				EXIT	TST	

2011 007636 104432
 2012 007640 000676
 2013
 2014 007642
 2015 007642
 2016 007642
 2017 007642 005237 007650
 2018 007646
 2019 007646
 2020 007646 000002
 2021
 2022 007650 000000
 2023
 2024 007652
 2025 007652
 2026 007652
 2027 007652 012746 010150
 2028 007656 012746 000001
 2029 007662 010600
 2030 007664 104414
 2031 007666 062706 000004
 2032 007672
 2033 007672 013746 007650
 2034 007676 012746 010224
 2035 007702 012746 000002
 2036 007706 010600
 2037 007710 104414
 2038 007712 062706 000006
 2039 007716
 2040 007716
 2041 007716 104423
 2042
 2043 007720
 2044 007720
 2045 007720
 2046 007720 012746 010273
 2047 007724 012746 000001
 2048 007730 010600
 2049 007732 104414
 2050 007734 062706 000004
 2051 007740
 2052 007740 013746 007650
 2053 007744 012746 010224
 2054 007750 012746 000002
 2055 007754 010600
 2056 007756 104414
 2057 007760 062706 000006
 2058 007764
 2059 007764
 2060 007764 104423
 2061
 2062 007766
 2063 007766
 2064 007766
 2065 007766 012746 010347
 2066 007772 012746 000001

```

TRAP C$EXIT
.WORD L10044-.

INTSR:
BGNSRV BEVENT ;INTERRUPT SERVICE ROUTINE
BEVENT::
INC ICOUNT ;INCREMENT COUNTER
ENDSRV
L10052:
RTI

ICOUNT: .WORD 0

BGNMSG BVERR1
BVERR1::
PRINTB #MSG1
MOV #MSG1,-(SP)
MOV #1,-(SP)
MOV SP,R0
TRAP C$PNTB
ADD #4,SP
PRINTB #INTCT,ICOUNT
MOV ICOUNT,-(SP)
MOV #INTCT,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C$PNTB
ADD #6,SP
ENDMSG
L10053:
TRAP C$MSG

BGNMSG BVERR2
BVERR2::
PRINTB #MSG2
MOV #MSG2,-(SP)
MOV #1,-(SP)
MOV SP,R0
TRAP C$PNTB
ADD #4,SP
PRINTB #INTCT,ICOUNT
MOV ICOUNT,-(SP)
MOV #INTCT,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C$PNTB
ADD #6,SP
ENDMSG
L10054:
TRAP C$MSG

BGNMSG BVERR3
BVERR3::
PRINTB #MSG3
MOV #MSG3,-(SP)
MOV #1,-(SP)
  
```

2067	007776	010600			MOV	SP,R0
2068	010000	104414			TRAP	C\$PNTB
2069	010002	062706	000004		ADD	#4,SP
2070	010006				PRINTB	#INTCT,ICOUNT
2071	010006	013746	007650		MOV	ICOUNT,-(SP)
2072	010012	012746	010224		MOV	#INTCT,-(SP)
2073	010016	012746	000002		MOV	#2,-(SP)
2074	010022	010600			MOV	SP,R0
2075	010024	104414			TRAP	C\$PNTB
2076	010026	062706	000006		ADD	#6,SP
2077	010032				ENDMSG	
2078	010032			L10055:		
2079	010032	104423			TRAP	C\$MSG
2080						
2081	010034				BGNMSG	BVERR4
2082	010034			BVERR4::		
2083	010034				PRINTB	#MSG4
2084	010034	012746	010414		MOV	#MSG4,-(SP)
2085	010040	012746	000001		MOV	#1,-(SP)
2086	010044	010600			MOV	SP,R0
2087	010046	104414			TRAP	C\$PNTB
2088	010050	062706	000004		ADD	#4,SP
2089	010054				PRINTB	#INTCT,ICOUNT
2090	010054	013746	007650		MOV	ICOUNT,-(SP)
2091	010060	012746	010224		MOV	#INTCT,-(SP)
2092	010064	012746	000002		MOV	#2,-(SP)
2093	010070	010600			MOV	SP,R0
2094	010072	104414			TRAP	C\$PNTB
2095	010074	062706	000006		ADD	#6,SP
2096	010100				ENDMSG	
2097	010100			L10056:		
2098	010100	104423			TRAP	C\$MSG
2099						
2100	010102				BGNMSG	BVERR5
2101	010102			BVERR5::		
2102	010102				PRINTB	#MSG5
2103	010102	012746	010464		MOV	#MSG5,-(SP)
2104	010106	012746	000001		MOV	#1,-(SP)
2105	010112	010600			MOV	SP,R0
2106	010114	104414			TRAP	C\$PNTB
2107	010116	062706	000004		ADD	#4,SP
2108	010122				PRINTB	#INTCT,ICOUNT
2109	010122	013746	007650		MOV	ICOUNT,-(SP)
2110	010126	012746	010224		MOV	#INTCT,-(SP)
2111	010132	012746	000002		MOV	#2,-(SP)
2112	010136	010600			MOV	SP,R0
2113	010140	104414			TRAP	C\$PNTB
2114	010142	062706	000006		ADD	#6,SP
2115	010146				ENDMSG	
2116	010146			L10057:		
2117	010146	104423			TRAP	C\$MSG
2118						
2119	010150	040445	042502	042526	MSG1:	.ASCIZ /%ABEVENT CLAMP FAILED TO ALLOW INTERRUPTS%N/
2120	010156	052116	041440	040514		
2121	010164	050115	043040	044501		
2122	010172	042514	020104	047524		

2123 010200 040440 046114 053517
2124 010206 044440 052116 051105
2125 010214 052522 052120 022523
2126 010222 000116
2127 010224 040445 052516 041115
2128 010232 051105 047440 020106
2129 010240 047111 042524 051122
2130 010246 050125 051524 051040
2131 010254 041505 044505 042526
2132 010262 035104 022440 031517
2133 010270 047045 000
2134 010273 045 041101 053105
2135 010300 047105 020124 046103
2136 010306 046501 020120 044504
2137 010314 020104 047516 020124
2138 010322 051120 053105 047105
2139 010330 020124 047111 042524
2140 010336 051122 050125 051524
2141 010344 047045 000
2142 010347 045 051101 051505
2143 010354 052105 042040 042111
2144 010362 047040 052117 050040
2145 010370 042522 042526 052116
2146 010376 044440 052116 051105
2147 010404 052522 052120 022523
2148 010412 000116
2149 010414 040445 051120 047511
2150 010422 044522 054524 032440
2151 010430 042040 042111 047040
2152 010436 052117 040440 046114
2153 010444 053517 044440 052116
2154 010452 051105 052522 052120
2155 010460 022523 000116
2156 010464 040445 051120 047511
2157 010472 044522 054524 033040
2158 010500 042040 042111 047040
2159 010506 052117 050040 042522
2160 010514 042526 052116 044440
2161 010522 052116 051105 052522
2162 010530 052120 022523 000116
2163
2164 010536
2165 010536
2166 010536 104401

INTCT: .ASCIZ /%ANUMBER OF INTERRUPTS RECEIVED: %03%N/

MSG2: .ASCIZ /%ABEVENT CLAMP DID NOT PREVENT INTERRUPTS%N/

MSG3: .ASCIZ /%ARESET DID NOT PREVENT INTERRUPTS%N/

MSG4: .ASCIZ /%APRIORITY 5 DID NOT ALLOW INTERRUPTS%N/

MSG5: .ASCIZ /%APRIORITY 6 DID NOT PREVENT INTERRUPTS%N/

L10044: .EVEN
ENDTST
TRAP C\$ETST

```

2167
2168
2169
2170
2171
2172
2173 010540          BGNTST
2174
2175 010540 005037 177524    CLR      LSREG      ;TURN ON ALL FOUR LED'S
2176 010544 004537 005212    JSR      R5,WDELAY ;DELAY APPROX. 200MS
2177 010550 000310
2178 010552          BREAK
2179 010552 104422          TRAP     C$BRK      ;CHECK SUPERVISOR FOR CONTROL REQUESTS
2180 010554 012737 000017 177524    MOV      #17,LSREG
2181 010562 004537 005212    JSR      R5,WDELAY ;TURN OFF ALL FOUR LED'S
2182 010566 000310          200.
2183 010570          MANUAL
2184 010570 104450          TRAP     C$MANI
2185 010572          BCOMPLETE 2$      ;BR IF YES
2186 010572 103410          BCS     2$
2187 010574 022737 000030 002266    CMP      #30,PASCT ;IS PASS COUNT >= 30?
2188 010602 003402          BLE     1$      ;BR IF YES
2189 010604
2190 010604 104432          EXIT    TST
2191 010606 000100          TRAP     C$EXIT
2192 010610 005037 002266 1$:      .WORD   L10060-.
2193 010614 012737 000016 177524 2$:      CLR     PASCT      ;EXIT TEST
2194 010622 004537 005212    MOV      #16,LSREG ;TURN ON THE LED CORRESPONDING TO THE LSB
2195 010626 000310          JSR      R5,WDELAY ;DELAY APPROX. 200MS
2196 010630          200.
2197 010630 104422          BREAK
2198 010632 012737 000015 177524    TRAP     C$BRK
2199 010640 004537 005212    MOV      #15,LSREG ;TURN ON 2ND LED
2200 010644 000310          JSR      R5,WDELAY ;DELAY APPROX 200 MS
2201 010646          200.
2202 010646 104422          BREAK
2203 010650 012737 000013 177524    TRAP     C$BRK
2204 010656 004537 005212    MOV      #13,LSREG ;TURN ON 3RD LED
2205 010662 000310          JSR      R5,WDELAY ;DELAY APPROX 200MS
2206 010664          200.
2207 010664 104422          BREAK
2208 010666 012737 000007 177524    TRAP     C$BRK
2209 010674 004537 005212    MOV      #7,LSREG  ;TURN ON LED CORRESPONDING TO MSB
2210 010700 000310          JSR      R5,WDELAY ;DELAY APPROX 200MS
2211 010702          200.
2212 010702 104432          EXIT    TST
2213 010704 000002          TRAP     C$EXIT
2214          .WORD   L10060-.
2215 010706          ENDTST
2216 010706          L10060:
2217 010706 104401          TRAP     C$ETST
2218

```



```

2219 .SBTTL TEST 5: ROCKER SWITCHES TEST
2220 :TEST TO CONFIRM THE ROCKER SWITCH SETTINGS. THIS TEST ASSUMES THAT,
2221 :IN MANUFACTURING, THE ROCKER SWITCHES ARE ALL IN THE ON POSITION.
2222 :THIS INCLUDES BOTH E21 AND E15 ON THE BDV11, OR SWITCH E102 ON THE KDF11-B. IN
2223 :MANUFACTURING, THIS TEST WILL VERIFY THAT ALL SWITCHES CAN BE READ AS ON. IN,
2224 :OTHER ENVIRONMENTS THE OPERATOR MAY SPECIFY WHAT THE SWITCH SETTINGS ARE BEFORE
2225 :THE DIAGNOSTIC IS STARTED (SEE PROGRAM OPTIONS UNDER OPERATING
2226 :INSTRUCTIONS). SWITCHES A1-A8 CORRESPOND TO E15 AND SWITCHES
2227 :B1-B4 TO E21 ON THE BDV11.
2228 010710 BGNTST
2229
2230 010710 MANUAL ;IS MANUAL INTERVENTION ALLOWED?
2231 010710 104450 TRAP C$MANI
2232 010712 BCOMPLETE PRTSW ;BR IF YES
2233 010712 103420 BCS PRTSW
2234 010714 005737 002240 TST KDF11B ;IF THIS IS A KDF11-B...
2235 010720 001402 BEQ 3$
2236 010722 105037 002277 CLR B SWSET+1 ;...THEN JUST LOOK AT 8 SWITCHES
2237 010726 023737 002276 177524 3$: CMP SWSET,LSREG ;ALL SWITCHES SHOULD BE ON & BITS 0-11..
2238 ;... (OR BITS 0-7 IF ON A KDF11-B)...
2239 ;... SHOULD BE SET.
2240 010734 001404 BEQ 1$ ;BR IF SWITCH READINGS ARE OK
2241 010736 ERRDF 27, SWERR ;CANNOT READ SWITCHES PROPERLY
2242 010736 104455 TRAP C$ERRDF
2243 010740 000033 .WORD 27
2244 010742 000000 .WORD 0
2245 010744 011306 .WORD SWERR
2246 010746 1$: CKLOOP ;CHECK FOR LOOP ON ERROR
2247 010746 104406 TRAP C$CLP1
2248 010750 EXIT TST ;EXIT
2249 010750 104432 TRAP C$EXIT
2250 010752 000566 .WORD L10061-
2251 010754 013737 177524 011302 PRTSW: MOV LSREG,TEMP ;COPY CONTENTS OF LSREG
2252 010762 005037 011304 CLF SWCHON ;CLEAR MASK
2253 010766 012737 000014 011300 MOV #14,SWCNT ;SET SWITCH COUNT
2254 010774 005737 002240 TST KDF11B ;IF THIS IS A KDF11-B...
2255 011000 001403 BEQ LP
2256 011002 012737 000010 011300 MOV #10,SWCNT ;...THEN JUST TEST FOR 8 SWITCHES
2257 011010 032737 000001 011302 LP: BIT #BIT0,TEMP ;TEST FOR SWITCH SET
2258 011016 001412 BEQ 2$ ;BR IF NOT SET
2259 011020 005737 002240 TST KDF11B ;SEE IF WE ARE ON A KDF11B
2260 011024 001404 BEQ 4$ ;BRANCH IF NOT
2261 011026 052737 004000 011304 BIS #BIT11,SWCHON ;SET CORRESPONDING BIT IN MASK
2262 011034 000403 BR 2$
2263 011036 052737 100000 011304 4$: BIS #BIT15,SWCHON ;IF SET, THEN SET CORRESPONDING BIT IN MASK
2264 011044 000241 2$: CLC ;CLEAR C-BIT FOR ROTATE
2265 011046 006037 011304 ROR SWCHON ;ROTATE SWSET
2266 011052 006037 011302 ROR TEMP ;GET READY TO TEST NEXT SWITCH
2267 011056 005337 011300 DEC SWCNT ;DECREMENT SWITCH COUNT
2268 011062 001352 BNE LP ;LOOP UNTIL ALL SWITCHES HAVE BEEN CHECKED
2269 011064 000241 CLC ;CLEAR C-BIT FOR ROTATE
2270 011066 006037 011304 ROR SWCHON ;ROTATE DATA
2271 011072 006037 011304 ROR SWCHON ;ROTATE DATA
2272 011076 006037 011304 ROR SWCHON ;ROTATE DATA
2273 011102 PRINTF #READN,SWCHON ;PRINT SWITCH SETTINGS
2274 011102 013746 011304 MOV SWCHON,-(SP)
  
```

```
2275 011106 012746 011340      MOV      #READN,-(SP)
2276 011112 012746 000002      MOV      #2,-(SP)
2277 011116 010600                MOV      SP,R0
2278 011120 104417                TRAP     C$PNTF
2279 011122 062706 000006      ADD      #6,SP
2280
2281 011126 013702 011304      MOV      SWCHON,R2
2282 011132 012701 000001      MOV      #1,R1
2283 011136 032702 000001      TAG1:   BIT      #BIT0,R2
2284 011142 001411                BEQ      TAG2
2285 011144                PRINTF   #MESSG1,R1
2286 011144 010146      MOV      R1,-(SP)
2287 011146 012746 011373      MOV      #MESSG1,-(SP)
2288 011152 012746 000002      MOV      #2,-(SP)
2289 011156 010600                MOV      SP,R0
2290 011160 104417                TRAP     C$PNTF
2291 011162 062706 000006      ADD      #6,SP
2292 011166 005201      TAG2:   INC      R1
2293 011170 006002                ROR      R2
2294 011172 022701 000010      CMP      #10,R1
2295 011176 002357                BGE      TAG1
2296 011200 005737 002240      TST      KDF11B
2297 011204 001023                BNE      TAG4A
2298 011206 012701 000001      MOV      #1,R1
2299 011212 032702 000001      TAG3:   BIT      #BIT0,R2
2300 011216 001411                BEQ      TAG4
2301 011220                PRINTF   #MESSG2,R1
2302 011220 010146      MOV      R1,-(SP)
2303 011222 012746 011406      MOV      #MESSG2,-(SP)
2304 011226 012746 000002      MOV      #2,-(SP)
2305 011232 010600                MOV      SP,R0
2306 011234 104417                TRAP     C$PNTF
2307 011236 062706 000006      ADD      #6,SP
2308 011242 005201      TAG4:   INC      R1
2309 011244 006002                ROR      R2
2310 011246 022701 000004      CMP      #4,R1
2311 011252 002357                BGE      TAG3
2312 011254      TAG4A: PRINTF   #NEWLIN
2313 011254 012746 011421      MOV      #NEWLIN,-(SP)
2314 011260 012746 000001      MOV      #1,-(SP)
2315 011264 010600                MOV      SP,R0
2316 011266 104417                TRAP     C$PNTF
2317 011270 062706 000004      ADD      #4,SP
2318
2319 011274                EXIT TST
2320 011274 104432      TRAP     C$EXIT
2321 011276 000242      .WORD    L10061-.
2322
2323 011300 000000      SWCNT:   .WORD    0
2324 011302 000000      TEMP:   .WORD    0
2325 011304 000000      SWCHON: .WORD    0
2326
2327 011306                BGNMSG   SWERR
2328 011306      SWERR::
2329 011306                PRINTB   #SERR1,SWSET,LSREG
2330 011306 013746 177524      MOV      LSREG,-(SP)
```

```
:COPY SWITCH SETTINGS
:SET SWITCH NUMBER = 1
:IS THIS SWITCH ON?
:BR IF NO
:PRINT SWITCH NUMBER

:INCREMENT SWITCH NUMBER
:ROTATE SWITCH REGISTER
:FINISHED WITH E15 ON BDV11 OR E102 ON KDF11-B?
:BR IF NO
:SEE IF WE ARE ON A KDF11-B
:BRANCH IF YES
:RESET SWITCH NUMBER FOR E21 ON BDV11
:IS THIS SWITCH SET?
:BR IF NO
:PRINT SWITCH NUMBER

:INCREMENT SWITCH NUMBER
:ROTATE SWITCH REGISTER
:FINISHED?
:BR IF NO
```

2331	011312	013746	002276		MOV	SWSET, -(SP)
2332	011316	012746	011424		MOV	#SERR1, -(SP)
2333	011322	012746	000003		MOV	#3, -(SP)
2334	011326	010600			MOV	SP, R0
2335	011330	104414			TRAP	C\$PNTB
2336	011332	062706	000010		ADD	#10, SP
2337	011336				ENDMSG	
2338	011336			L10062:		
2339	011336	104423			TRAP	C\$MSG
2340						
2341	011340	047045	040445	053523	READN:	.ASCIZ /%N%SWITCHES ON : %06%A : /
2342	011346	052111	044103	051505		
2343	011354	047440	020116	020072		
2344	011362	047445	022466	020101		
2345	011370	020072	000			
2346	011373	045	040501	042045	MESSG1:	.ASCIZ /%AA%D1%A, /
2347	011400	022461	026101	000040		
2348	011406	040445	022502	030504	MESSG2:	.ASCIZ /%AB%D1%A, /
2349	011414	040445	020054	000		
2350	011421	045	000116		NEWLIN:	.ASCIZ /%N/
2351	011424	040445	044504	020104	SERR1:	.ASCII /%ADID NOT RECOGNIZE ALL SWITCHES AS ON%N/
2352	011432	047516	020124	042522		
2353	011440	047503	047107	055111		
2354	011446	020105	046101	020114		
2355	011454	053523	052111	044103		
2356	011462	051505	040440	020123		
2357	011470	047117	047045			
2358	011474	040445	054105	042520		.ASCIZ /%AEXPECTED: %06%S5%ARECEIVED: %06%N/
2359	011502	052103	042105	020072		
2360	011510	047445	022466	032523		
2361	011516	040445	042522	042503		
2362	011524	053111	042105	022472		
2363	011532	033117	047045	000		
2364		011540				
2365	011540				.EVEN	
2366	011540			L10061:	ENDTST	
2367	011540	104401			TRAP	C\$ETST
2368						

```

2369 .SBTTL TEST 6: 2K DIAGNOSTIC ROM
2370 :++
2371 :TEST TO PERFORM CHECKSUM AND CHECKWORD VERIFICATION ON THE 2K
2372 :OF DIAGNOSTIC ROM. IN UNATTENDED MODE, THE ROM WILL BE ADDRESSED
2373 :FROM 0-2K. IN STAND-ALONE MODE, THE OPERATOR MAY CHANGE THE
2374 :ADDRESS BY RESPONDING TO QUESTIONS GENERATED ON THE FIRST PASS.
2375 :--
2376 011542 BGNTST
2377
2378 011542 BGNSUB
2379 011542 104402 TRAP C$BSUB
2380 011544 MANUAL ;MANUAL INTERVENTION OK?
2381 011544 104450 TRAP C$MANI
2382 011546 BNCOMPLETE STRT ;BR IF NO
2383 011546 103014 BCC STRT
2384 011550 005737 002264 TST PASS ;FIRST PASS?
2385 011554 001032 BNE RSTRT ;BR IF NO
2386 011556 GMANIL DADDR,RSET,1,YES
2387 011556 104443 TRAP C$GMAN
2388 011560 000404 BR 10000$
2389 011562 002314 .WORD RSET
2390 011564 000130 .WORD T$CODE
2391 011566 013622 .WORD DADDR
2392 011570 000001 .WORD 1
2393 011572 10000$:
2394 011572 005737 002314 TST RSET ;STANDARD JUMPERS?
2395 011576 001404 BEQ GETAD ;BR IF NO
2396 011600 012737 000400 012472 STRT: MOV #400,DRLP ;STORE STARTING ADDRESS
2397 011606 000415 BR RSTRT ;GO PERFORM TEST
2398 011610 GETAD: GMANID LOADR,STORE,D,-1,0,24,NO
2399 011610 104443 TRAP C$GMAN
2400 011612 000406 BR 10001$
2401 011614 002300 .WORD STORE
2402 011616 000042 .WORD T$CODE
2403 011620 003052 .WORD LOADR
2404 011622 177777 .WORD -1
2405 011624 000000 .WORD T$LOLIM
2406 011626 000024 .WORD T$HILIM
2407 011630 10001$:
2408 011630 004737 005122 JSR PC,SETADR ;GET STARTING ADDRESS
2409 011634 013737 002250 012472 MOV LOPAG,DRLP ;STORE STARTING ADDRESS
2410 011642 013737 012472 002242 RSTRT: MOV DRLP,VRTPCR ;SET UP PCR
2411 011650 013737 002242 177520 MOV VRTPCR,PCR
2412 011656 012737 000010 002252 DRTST: MOV #10,COUNTR ;SET NUMBER OF CHECKWORDS TO CHECK
2413 011664 012705 002160 MOV #SFPTBL,R5 ;LOCATION OF CHECKWORDS
2414 011670 012737 000001 002256 MOV #1,RFLAG ;INDICATE ROM
2415 011676 005037 002244 DLOOP: CLR BCF ;SIGNAL LOW BYTES ARE BEING CHECKED
2416 011702 122737 177777 173774 CMPB #-1,@#173774 ;DOES THE ROM EXIST?
2417 011710 001007 BNE 1$ ;BR IF YES
2418 011712 ERRDF 30,,DERR1 ;LOW BYTE DIAGNOSTIC ROM NOT FOUND
2419 011712 104455 TRAP C$ERRDF
2420 011714 000036 .WORD 30
2421 011716 000000 .WORD 0
2422 011720 012474 .WORD DERR1
2423 011722 CKLOOP ;LOOP ON ERROR IF SELECTED
2424 011722 104406 TRAP C$CLP1

```

2425	011724				EXIT	TST			:EXIT TEST,ROM NOT FOUND
2426	011724	104432			TRAP	C\$EXIT			
2427	011726	001716			.WORD	L10063-			
2428	011730			1\$:	CKLOOP				:CHECK FOR LOOP ON ERROR
2429	011730	104406			TRAP	C\$CLP1			
2430	011732	004737	004260		JSR	PC,CHKSUM			:COMPUTE THE ACTUAL CHECKSUM
2431	011736	113737	173776	002260	MOVB	@#173776,EXPSUM			:GET THE STORED CHECKSUM
2432	011744	063737	002262	002260	ADD	ACTSUM,EXPSUM			:ADD THE EXPECTED AND ACTUAL CHECKSUMS
2433	011752	105737	002260		TSTB	EXPSUM			:BYTE RESULT = 0?
2434	011756	001404			BEQ	2\$:BR IF YES
2435	011760				ERRDF	31,,DERR2			:CHECKSUM ERROR IN DIAGNOSTIC ROM
2436	011760	104455			TRAP	C\$ERDF			
2437	011762	000037			.WORD	31			
2438	011764	000000			.WORD	0			
2439	011766	012522			.WORD	DERR2			
2440	011770			2\$:	CKLOOP				:CHECK FOR LOOP ON ERROR
2441	011770	104406			TRAP	C\$CLP1			
2442	011772				ENDSUB				
2443	011772			L10064:					
2444	011772	104403			TRAP	C\$ESUB			
2445									
2446									
2447	011774				BGNSUB				
2448	011774	104402			TRAP	C\$BSUB			
2449	011776	012737	000001	002244	MOV	#1,BCF			:SET BCF TO DENOTE HIGH BYTES
2450	012004	122737	177777	173775	CMPB	#-1,@#173775			:DOES THE ROM EXIST?
2451	012012	001007			BNE	3\$:BR IF YES
2452	012014				ERRDF	32,,DERR3			:HIGH BYTE DIAGNOSTIC ROM NOT FOUND
2453	012014	104455			TRAP	C\$ERDF			
2454	012016	000040			.WORD	32			
2455	012020	000000			.WORD	0			
2456	012022	012550			.WORD	DERR3			
2457	012024				CKLOOP				:LOOP ON ERROR IF SELECTED
2458	012024	104406			TRAP	C\$CLP1			
2459	012026				EXIT	TST			:EXIT TEST, ROM NOT FOUND
2460	012026	104432			TRAP	C\$EXIT			
2461	012030	001614			.WORD	L10063-			
2462	012032			3\$:	CKLOOP				:CHECK FOR LOOP ON ERROR
2463	012032	104406			TRAP	C\$CLP1			
2464	012034	004737	004260		JSR	PC,CHKSUM			:COMPUTE THE ACTUAL CHECKSUM
2465	012040	113737	173777	002260	MOVB	@#173777,EXPSUM			:GET EXPECTED CHECKSUM
2466	012046	063737	002262	002260	ADD	ACTSUM,EXPSUM			:ADD THE EXPECTED AND ACTUAL CHECKSUMS
2467	012054	105737	002260		TSTB	EXPSUM			:BYTE RESULT = 0?
2468	012060	001404			BEQ	4\$:BR IF YES
2469	012062				ERRDF	33,,DERR4			:CHECKSUM ERROR IN DIAGNOSTIC ROM
2470	012062	104455			TRAP	C\$ERDF			
2471	012064	000041			.WORD	33			
2472	012066	000000			.WORD	0			
2473	012070	012576			.WORD	DERR4			
2474	012072			4\$:	CKLOOP				
2475	012072	104406			TRAP	C\$CLP1			
2476	012074				ENDSUB				
2477	012074			L10065:					
2478	012074	104403			TRAP	C\$ESUB			
2479									
2480	012076				BGNSUB				

```

2481 012076 104402          TRAP    C$BSUB
2482 012100 062737 001002 002242  ADD    #1002,VRTPCR          ;NEXT PAGE IN PCR
2483 012106 013737 002242 177520  MOV    VRTPCR,PCR
2484 012114 005337 002252          DEC    COUNTR              ;DECREMENT CHECKWORD COUNT
2485 012120 001266          BNE    DLOOP                ;LOOP UNTIL ALL 20 PAGES HAVE BEEN CHECKED
2486 012122          ENDSUB
2487 012122          L10066: TRAP    C$ESUB
2488 012122 104403
2489
2490
2491          ;GET THE CHECKWORDS FROM THE ROMS AND PUT INTO TABLE 'CHKWRD'
2492 012124          BGNSUB
2493 012124 104402          TRAP    C$BSUB
2494 012126 012702 012652  MOV    #CHKWRD,R2
2495 012132 012737 000001 002242  MOV    #1,VRTPCR
2496 012140 012737 000010 002252  MOV    #10,COUNTR
2497 012146 013737 002242 177520 5$:   MOV    VRTPCR,PCR
2498 012154 013722 173376  MOV    @#173376,(R2)+
2499 012160 062737 000002 002242  ADD    #2,VRTPCR
2500 012166 005337 002252          DEC    COUNTR
2501 012172 001365          BNE    5$
2502 012174          ENDSUB
2503 012174          L10067: TRAP    C$ESUB
2504 012174 104403
2505
2506
2507          ;TRY TO IDENTIFY THE ROM CHIPS
2508 012176          BGNSUB
2509 012176 104402          TRAP    C$BSUB
2510 012200 013701 012672  MOV    TABLES,R1
2511 012204 012737 002160 012674  MOV    #SFPTBL,PNTR
2512 012212 013700 012674 6$:   MOV    PNTR,R0
2513 012216 012702 012652  MOV    #CHKWRD,R2
2514 012222 012737 000010 002252  MOV    #10,COUNTR
2515 012230 022022 7$:   CMP    (R0)+,(R2)+          ;ARE THE CHECKWORDS EQUAL?
2516 012232 001047          BNE    9$                    ;BRANCH IF NOT
2517 012234 005337 002252  DEC    COUNTR                ;DONE CHECKING THIS TABLE?
2518 012240 001373          BNE    7$                    ;BRANCH IF NOT
2519 012242 020127 000003  CMP    R1,#3                ;DID THE FIRST TABLE OF CHECKWORDS COMPARE?
2520 012246 001011          BNE    8$                    ;BRANCH IF NOT
2521 012250          PRINTF #TABL1              ;PRINT OUT ROM CHIP NUMBERS
2522 012250 012746 013134  MOV    #TABL1,-(SP)
2523 012254 012746 000001  MOV    #1,-(SP)
2524 012260 010600          MOV    SP,R0
2525 012262 104417          TRAP    C$PNTF
2526 012264 062706 000004  ADD    #4,SP
2527 012270 000475          BR     11$
2528 012272 020127 000002 8$:   CMP    R1,#2                ;DID THE SECOND TABLE OF CHECKWORDS COMPARE?
2529 012276 001011          BNE    20$                   ;BRANCH IF NOT
2530 012300          PRINTF #TABL2              ;PRINT OUT ROM CHIP NUMBERS
2531 012300 012746 013233  MOV    #TABL2,-(SP)
2532 012304 012746 000001  MOV    #1,-(SP)
2533 012310 010600          MOV    SP,R0
2534 012312 104417          TRAP    C$PNTF
2535 012314 062706 000004  ADD    #4,SP
2536 012320 000461          BR     11$

```

```
2537 012322 020127 000001      20$:  CMP      R1,#1
2538 012326 001011              BNE      9$
2539 012330                    PRINTF   #TABL3
2540 012330 012746 013332      MOV      #TABL3,-(SP)
2541 012334 012746 000001      MOV      #1,-(SP)
2542 012340 010600              MOV      SP,R0
2543 012342 104417              TRAP     C$PNTF
2544 012344 062706 000004      ADD      #4,SP
2545 012350 000445              BR       11$
2546 012352 062737 000020 012674 9$:  ADD      #20,PNTR
2547 012360 005301              DEC      R1
2548 012362 001313              BNE      6$
2549 012364                    MANUAL
2550 012364 104450              TRAP     C$MANI
2551 012366                    BCOMPLETE 100$
2552 012366 103405              BCS     100$
2553 012370                    ERRDF    34,,DERR5
2554 012370 104455              TRAP     C$ERDF
2555 012372 000042              .WORD   34
2556 012374 000000              .WORD   0
2557 012376 012624              .WORD   DERR5
2558 012400                    CKLOOP
2559 012400 104406              TRAP     C$CLP1
2560 012402                    PRINTF   #UNKNWN
2561 012402 012746 013431 100$:  MOV      #UNKNWN,-(SP)
2562 012406 012746 000001      MOV      #1,-(SP)
2563 012412 010600              MOV      SP,R0
2564 012414 104417              TRAP     C$PNTF
2565 012416 062706 000004      ADD      #4,SP
2566 012422 012737 000010 002252 MOV      #10,COUNTR
2567 012430 012701 012652      MOV      #CHKWRD,R1
2568 012434                    PRINTF   #CHECKW,(R1)+
2569 012434 012146      MOV      (R1)+,-(SP)
2570 012436 012746 013515      MOV      #CHECKW,-(SP)
2571 012442 012746 000002      MOV      #2,-(SP)
2572 012446 010600              MOV      SP,R0
2573 012450 104417              TRAP     C$PNTF
2574 012452 062706 000006      ADD      #6,SP
2575 012456 005337 002252      DEC      COUNTR
2576 012462 001364              BNE      10$
2577 012464                    11$:  ENDSUB
2578 012464                    L10070:
2579 012464 104403              TRAP     C$ESUB
2580
2581
2582
2583 012466                    EXIT     TST
2584 012466 104432              TRAP     C$EXIT
2585 012470 001154              .WORD   L10063-.
2586
2587 012472 000000              DRLP:   .WORD   0
2588
2589 012474                    BGNMSG  DERR1
2590 012474                    DERR1::
2591 012474                    PRINTB  #LRAERR,#NODR
2592 012474 012746 013576      MOV      #NODR,-(SP)
```

```
;ANY MORE TABLES TO CHECK?
;BRANCH IF YES
;APT MODE?
;PRINT UNKNOWN
;CHECKWORD ERROR
;ROMS DO NOT MATCH
;PRINT OUT UNIDENTIFIED ROM CHIPS CHECKWORDS
```

2593	012500	012746	012676	MOV	#LRAERR,-(SP)
2594	012504	012746	000002	MOV	#2,-(SP)
2595	012510	010600		MOV	SP,R0
2596	012512	104414		TRAP	C\$PNTB
2597	012514	062706	000006	ADD	#6,SP
2598	012520			ENDMSG	
2599	012520			L10071:	
2600	012520	104423		TRAP	C\$MSG
2601					
2602	012522			BGNMSG	DERR2
2603	012522			DERR2::	
2604	012522			PRINTB	#LOWROM,#CKERR
2605	012522	012746	002702	MOV	#CKERR,-(SP)
2606	012526	012746	012754	MOV	#LOWROM,-(SP)
2607	012532	012746	000002	MOV	#2,-(SP)
2608	012536	010600		MOV	SP,R0
2609	012540	104414		TRAP	C\$PNTB
2610	012542	062706	000006	ADD	#6,SP
2611	012546			ENDMSG	
2612	012546			L10072:	
2613	012546	104423		TRAP	C\$MSG
2614					
2615	012550			BGNMSG	DERR3
2616	012550			DERR3::	
2617	012550			PRINTB	#HRAERR,#NODR
2618	012550	012746	013576	MOV	#NODR,-(SP)
2619	012554	012746	013014	MOV	#HRAERR,-(SP)
2620	012560	012746	000002	MOV	#2,-(SP)
2621	012564	010600		MOV	SP,R0
2622	012566	104414		TRAP	C\$PNTB
2623	012570	062706	000006	ADD	#6,SP
2624	012574			ENDMSG	
2625	012574			L10073:	
2626	012574	104423		TRAP	C\$MSG
2627					
2628	012576			BGNMSG	DERR4
2629	012576			DERR4::	
2630	012576			PRINTB	#HIROM,#CKERR
2631	012576	012746	002702	MOV	#CKERR,-(SP)
2632	012602	012746	013073	MOV	#HIROM,-(SP)
2633	012606	012746	000002	MOV	#2,-(SP)
2634	012612	010600		MOV	SP,R0
2635	012614	104414		TRAP	C\$PNTB
2636	012616	062706	000006	ADD	#6,SP
2637	012622			ENDMSG	
2638	012622			L10074:	
2639	012622	104423		TRAP	C\$MSG
2640					
2641	012624			BGNMSG	DERR5
2642	012624			DERR5::	
2643	012624			PRINTB	#MISTAK
2644	012624	012746	013524	MOV	#MISTAK,-(SP)
2645	012630	012746	000001	MOV	#1,-(SP)
2646	012634	010600		MOV	SP,R0
2647	012636	104414		TRAP	C\$PNTB
2648	012640	062706	000004	ADD	#4,SP

HARDWARE TESTS MACY11 30(1046) 20-JUL-81 11:44 PAGE 56
 CVMBAD.P11 10-JUL-81 17:08 TEST 6: 2K DIAGNOSTIC ROM

SEQ 0056

```

2649 012644 004737 004504 JSR PC,VIRTAD
2650 012650 ENDMMSG
2651 012650 L10075:
2652 012650 104423 TRAP C$MSG
2653
2654 012652 000010 CHKWRD: .BLKW 10 ;TABLE TO STORE THE CHECKWORDS
2655
2656 012672 000003 TABLES: .WORD 3 ;NUMBER OF CHECKWORD TABLES
2657 012674 000000 PNTR: .WORD 0 ;WILL BE USED AS A POINTER
2658
2659
2660 012676 052045 047045 040445 LRAERR: .ASCIZ /%T%N%ACANNOT ACCESS LOW BYTE DIAGNOSTIC ROM%/
2661 012704 040503 047116 052117
2662 012712 040440 041503 051505
2663 012720 020123 047514 020127
2664 012726 054502 042524 042040
2665 012734 040511 047107 051517
2666 012742 044524 020103 047522
2667 012750 022515 000116
2668
2669 012754 052045 047045 040445 LOWROM: .ASCIZ /%T%N%ALOW BYTE DIAGNOSTIC ROM%/
2670 012762 047514 020127 054502
2671 012770 042524 042040 040511
2672 012776 047107 051517 044524
2673 013004 020103 047522 022515
2674 013012 000116
2675
2676 013014 052045 047045 040445 HRAERR: .ASCIZ /%T%N%ACANNOT ACCESS HIGH BYTE DIAGNOSTIC ROM%/
2677 013022 040503 047116 052117
2678 013030 040440 041503 051505
2679 013036 020123 044510 044107
2680 013044 041040 052131 020105
2681 013052 044504 043501 047516
2682 013060 052123 041511 051040
2683 013066 046517 047045 000
2684
2685 013073 045 022524 022516 HIROM: .ASCIZ /%T%N%AHIGH BYTE DIAGNOSTIC ROM%/
2686 013100 044101 043511 020110
2687 013106 054502 042524 042040
2688 013114 040511 047107 051517
2689 013122 044524 020103 047522
2690 013130 022515 000116
2691
2692 013134 047045 040445 044103 TABL1: .ASCIZ /%N%ACHECKWORDS CORRESPOND TO ROM CHIPS #23-045E2 & #23-046E2%/
2693 013142 041505 053513 051117
2694 013150 051504 041440 051117
2695 013156 042522 050123 047117
2696 013164 020104 047524 051040
2697 013172 046517 041440 044510
2698 013200 051520 021440 031462
2699 013206 030055 032464 031105
2700 013214 023040 021440 031462
2701 013222 030055 033064 031105
2702 013230 047045 000
2703
2704 013233 045 022516 041501 TABL2: .ASCIZ /%N%ACHECKWORDS CORRESPOND TO ROM CHIPS #23-010E2 & #23-011E2%/

```

2705	013240	042510	045503	047527
2706	013246	042122	020123	047503
2707	013254	051122	051505	047520
2708	013262	042116	052040	020117
2709	013270	047522	020115	044103
2710	013276	050111	020123	031043
2711	013304	026463	030460	042460
2712	013312	020062	020046	031043
2713	013320	026463	030460	042461
2714	013326	022462	000116	
2715				
2716	013332	047045	040445	044103
2717	013340	041505	053513	051117
2718	013346	051504	041440	051117
2719	013354	042522	050123	047117
2720	013362	020104	047524	051040
2721	013370	046517	041440	044510
2722	013376	051520	021440	031462
2723	013404	031055	032064	031105
2724	013412	023040	021440	031462
2725	013420	031055	032464	031105
2726	013426	047045	000	
2727				
2728	013431	045	022516	052501
2729	013436	045516	047516	047127
2730	013444	051040	046517	041440
2731	013452	044510	051520	043040
2732	013460	052517	042116	020056
2733	013466	052040	042510	051111
2734	013474	041440	042510	045503
2735	013502	047527	042122	020123
2736	013510	051101	035105	000
2737				
2738	013515	045	022516	030517
2739	013522	000062		
2740				
2741	013524	040445	047111	047503
2742	013532	051122	041505	020124
2743	013540	044103	041505	053513
2744	013546	051117	020104	047111
2745	013554	042040	040511	047107
2746	013562	051517	044524	020103
2747	013570	047522	022515	000116
2748				
2749	013576	047516	026516	054105
2750	013604	051511	042524	052116
2751	013612	046440	046505	051117
2752	013620	000131		
2753				
2754	013622	052123	047101	040504
2755	013630	042122	045040	046525
2756	013636	042520	051522	000
2757				
2758		013644		
2759	013644			
2760	013644			

TABL3: .ASCIZ /%N%ACHECKWORDS CORRESPOND TO ROM CHIPS #23-244E2 & #23-245E2%N/

UNKNWN: .ASCIZ /%N%AUNKNOWN ROM CHIPS FOUND. THEIR CHECKWORDS ARE:/

CHECKW: .ASCIZ /%N%012/

MISTAK: .ASCIZ /%AINCORRECT CHECKWORD IN DIAGNOSTIC ROM%N/

NODR: .ASCIZ /NON-EXISTENT MEMORY/

DADDR: .ASCIZ /STANDARD JUMPERS/

.EVEN
 ENDTST
 L10063:

2761 013644 104401

2762
2763
2764
2765
2766
2767
2768
2769
2770
2771
2772
2773
2774
2775
2776
2777
2778
2779
2780
2781
2782
2783
2784
2785
2786 013646
2787
2788 013646
2789 013646 104402
2790 013650 005737 002240
2791 013654 001402
2792 013656
2793 013656 104432
2794 013660 002702
2795 013662
2796 013662 104450
2797 013664
2798 013664 103112
2799 013666 005037 015742
2800 013672 005737 002264
2801 013676 001422
2802 013700 005737 015746
2803 013704 001153
2804 013706 005737 015750
2805 013712 001402
2806 013714 000137 014610
2807 013720 005737 015752
2808 013724 001402
2809 013726 000137 015164
2810 013732 005737 015754
2811 013736 001465
2812 013740 000137 015530
2813 013744
2814 013744 104443
2815 013746 000404
2816 013750 015742

```

TRAP C$ETST
.SBTTL TEST 7: TEST ALL ADDITIONAL MEMORY
:++
:TEST TO LOCATE AND VERIFY CHECKSUMS IN ALL RESIDENT MEMORY
:ON A PAGE BASIS. THERE ARE FOUR STORAGE AREAS ASSOCIATED
:WITH THIS TEST WHICH HOLD THE CHECKWORDS OF ALL THE MEMORY
:THAT IS TO BE TESTED. THESE TABLES WILL HAVE DEFAULT VALUES
:ONLY IF THE ASSOCIATED MEMORY CHIP IS A STANDARD COMPONENT
:ON THE BOARD. IF NO DEFAULT VALUES EXIST, THE OPERATOR MUST
:INPUT THE CHECKWORDS AS LISTED ON THE PRINT SET. THE MEMORY
:WILL BE TESTED IN THE FOLLOWING LOCATIONS BY DEFAULT:
:
:   EXPANDABLE DIAGNOSTIC ROM      2-4K
:   EPROM IN SOCKETS              4-6K
:   SYSTEM ROM                     16-32K
:   SYSTEM EPROM                   16-24K
:
:THE TEST WILL FIRST VERIFY THE CHECKSUMS IN ALL RESIDENT ROM,
:THEN COMPARE THE ACTUAL CHECKWORDS. ERROR INFORMATION WILL
:INCLUDE THE SPECIFIC TYPE OF ERROR THAT OCCURS, THE VIRTUAL
:ADDRESS, AND WHETHER IT WAS THE HIGH BYTE OR LOW BYTE ROM/EPROM.
:THIS INFORMATION SHOULD ALLOW A KNOWLEDGEABLE OPERATOR TO ISOLATE
:THE ERROR DOWN TO A SINGLE ROM/EPROM WITH THE AID OF THE
:ADDRESS MAP IN THE PRINT SET.
:--

```

```

BGNTST
BGNSUB
TRAP C$BSUB
TST KDF11B ;IF THIS IS A KDF11-B...
BEQ 1$ ;...THEN SKIP THIS TEST
EXIT TST
TRAP C$EXIT
.WORD L10076-.
1$: MANUAL ;UNDER APT?
TRAP C$MANI
BNCOMPLETE DFLTST ;SKIP TEST IF YES
BCC DFLTST
CLR ADDON ;RESTORE DEFAULT
TST PASS ;FIRST PASS?
BEQ GET ;BR IF YES
TST LOD1 ;EXPANDED DIAGNOSTIC ROM?
BNE LD1 ;BR IF YES
TST LOD2 ;EPROM IN SOCKETS?
BEQ P1 ;BR IF NO
JMP LD2 ;TEST EPROM
P1: TST LOD3 ;SYSTEM ROM ?
BEQ P2 ;BR IF NO
JMP LD3 ;TEST ROM
P2: TST LOD4 ;SYSTEM EPROM?
BEQ DFLTST ;EXIT IF NO
JMP LD4 ;TEST EPROM
GET: GMANIL EXEC,ADDON,1,YES
TRAP C$GMAN
BR 10000$
.WORD ADDON

```

2817	013752	000130		.WORD	T\$CODE	
2818	013754	016303		.WORD	EXEC	
2819	013756	000001		.WORD	1	
2820	013760		10000\$:			
2821	013760	005737	015742	TST	ADDON	:ADDITIONAL MEMORY?
2822	013764	001452		BEQ	DFLTST	:BR IF NO
2823	013766		DIAIN:	GMANIL	EXPND,RESPND,1,NO	
2824	013766	104443		TRAP	C\$GMAN	
2825	013770	000404		BR	10001\$	
2826	013772	002312		.WORD	RESPND	
2827	013774	000120		.WORD	T\$CODE	
2828	013776	016332		.WORD	EXPND	
2829	014000	000001		.WORD	1	
2830	014002		10001\$:			
2831	014002	005737	002312	TST	RESPND	:EXPANDED DIAGNOSTIC ROM?
2832	014006	001045		BNE	EXPRCM	:BR IF YES
2833	014010		EPRIN:	GMANIL	EPRM,RESPND,1,NO	
2834	014010	104443		TRAP	C\$GMAN	
2835	014012	000404		BR	10002\$	
2836	014014	002312		.WORD	RESPND	
2837	014016	000120		.WORD	T\$CODE	
2838	014020	016362		.WORD	EPRM	
2839	014022	000001		.WORD	1	
2840	014024		10002\$:			
2841	014024	005737	002312	TST	RESPND	:EPROM IN SOCKETS?
2842	014030	001402		BEQ	SYSRIN	:BR IF NO
2843	014032	000137	014456	JMP	EPRMT	:JUMP TO ACCEPT INPUT
2844	014036		SYSRIN:	GMANIL	SYSR,RESPND,1,NO	
2845	014036	104443		TRAP	C\$GMAN	
2846	014040	000404		BR	10003\$	
2847	014042	002312		.WORD	RESPND	
2848	014044	000120		.WORD	T\$CODE	
2849	014046	016403		.WORD	SYSR	
2850	014050	000001		.WORD	1	
2851	014052		10003\$:			
2852	014052	005737	002312	TST	RESPND	:SYSTEM ROM?
2853	014056	001402		BEQ	SYSEIN	:BR IF NO
2854	014060	000137	015024	JMP	SYSRT	:INPUT CHECKWORDS
2855	014064		SYSEIN:	GMANIL	SYSE,RESPND,1,NO	
2856	014064	104443		TRAP	C\$GMAN	
2857	014066	000404		BR	10004\$	
2858	014070	002312		.WORD	RESPND	
2859	014072	000120		.WORD	T\$CODE	
2860	014074	016416		.WORD	SYSE	
2861	014076	000001		.WORD	1	
2862	014100		10004\$:			
2863	014100	005737	002312	TST	RESPND	:SYSTEM EPROM?
2864	014104	001402		BEQ	DFLTST	:BR IF NO
2865	014106	000137	015370	JMP	SYSET	:INPUT CHECKWORDS
2866	014112		DFLTST:	EXIT	TST	:NO ADDTL. MEMORY -- EXIT
2867	014112	104432		TRAP	C\$EXIT	
2868	014114	002446		.WORD	L10076-	
2869	014116		L10077:	ENDSUB		
2870	014116					
2871	014116	104403		TRAP	C\$ESUB	
2872						

2873							
2874	014120						
2875	014120	104402					
2876	014122	005037	002324				
2877	014126	012737	000010	002302	EXPROM:	CLR	C\$BSUB
2878	014134	012702	002330			MOV	ERRFLG
2879	014140	004737	004324			MOV	#10,WORDCT
2880	014144					MOV	#EXPDIA,R2
2881	014144	104443				JSR	PC,INPUT
2882	014146	000404				GMANIL	EXADD,ANSR,1,YES
2883	014150	002254				TRAP	C\$GMAN
2884	014152	000130				BR	10000\$
2885	014154	016433				.WORD	ANSR
2886	014156	000001				.WORD	T\$CODE
2887	014160					.WORD	EXADD
2888	014160	005737	002254		10000\$:	.WORD	1
2889	014164	001020				TST	ANSR
2890	014166	005237	002254			BNE	1\$
2891	014172					INC	ANSR
2892	014172	104443				GMANID	LOADR,STORE,D,-1,0,30,NO
2893	014174	000406				TRAP	C\$GMAN
2894	014176	002300				BR	10001\$
2895	014200	000042				.WORD	STORE
2896	014202	003052				.WORD	T\$CODE
2897	014204	177777				.WORD	LOADR
2898	014206	000000				.WORD	-1
2899	014210	000030				.WORD	T\$LOLIM
2900	014212					.WORD	T\$HILIM
2901	014212	004737	005122		10001\$:	JSR	PC,SETADR
2902	014216	013737	002250	015746		MOV	LOPAG,LOD1
2903	014224	000403				BR	LD1
2904	014226	012737	010420	015746	1\$:	MOV	#010420,LOD1
2905	014234	013737	015746	177520	LD1:	MOV	LOD1,PCR
2906	014242	012737	000001	002256		MOV	#1,RFLAG
2907	014250	012703	002330			MOV	#EXPDIA,R3
2908	014254	012737	000010	002252		MOV	#10,COUNTR
2909	014262	012337	002306		EXPTST:	MOV	(R3)+,CKWD
2910	014266	004737	004644			JSR	PC,MEMTST
2911	014272	005737	002246			TST	REAL
2912	014276	001457				BEQ	E3
2913	014300	005737	002324			TST	ERRFLG
2914	014304	001421				BEQ	NOERR
2915	014306	004737	004504			JSR	PC,VIRTAD
2916	014312	005737	002244			TST	BCF
2917	014316	001004				BNE	HIGH
2918	014320	012737	002745	002322		MOV	#LOBYT,BYTLOC
2919	014326	000403				BR	DATOUT
2920	014330	012737	003007	002322	HIGH:	MOV	#HIBYT,BYTLOC
2921	014336	022737	000001	002324	DATOUT:	CMP	#1,ERRFLG
2922	014344	001420				BEQ	E1
2923	014346	000425				BR	E2
2924	014350	062737	001002	177520	NOERR:	ADD	#1002,PCR
2925	014356	005337	002252			DEC	COUNTR
2926	014362	001337				BNE	EXPTST
2927	014364	005737	002264		MORE:	TST	PASS
2928	014370	001002				BNE	1\$

: CLEAR ERROR FLAG
: COUNT 8 CHECKWORDS
: POINTER TO STORAGE TABLE
: INPUT CHECKWORDS

: STANDARD MEMORY RANGE?
: BR IF YES
: RESTORE DEFAULT VALUE

: GET FIRST PAGE ADDRESS
: STORE LOW PAGE NO.
: SKIP NEXT INSTRUCTION
: STANDARD PAGE = 20,21 2-4K RANGE
: LOAD STARTING PAGE
: INDICATE ROM
: POINTER TO CHECKWORDS
: PAGE COUNT
: GET CHECKWORD FOR THIS PAGE
: TEST MEMORY
: DOES THE MEMORY EXIST?
: BR IF NO
: ANY OTHER ERRORS?
: BR IF NO
: GET ADDRESS OF ERROR
: LOW BYTE PAGE?
: BR IF NO
: SET POINTER FOR ERROR MSG.
: PRINT ERROR MESSAGE
: POINTER FOR ERROR MSG.
: CHECKSUM ERROR?
: BR IF YES
: ELSE CHECKWORD ERROR
: ADJUST PCR
: DEC PAGE COUNT
: LOOP UNTIL ALL PAGES ARE TESTED
: FIRST PASS?
: BR IF NO

```

2929 014372 000137 014010
2930 014376 000137 013720
2931 014402
2932 014402 104432
2933 014404 000046
2934 014406
2935 014406 104455
2936 014410 000043
2937 014412 016332
2938 014414 015760
2939 014416
2940 014416 104406
2941 014420 000761
2942 014422
2943 014422 104455
2944 014424 000044
2945 014426 016332
2946 014430 016016
2947 014432
2948 014432 104406
2949 014434 000753
2950 014436
2951 014436 104455
2952 014440 000045
2953 014442 016332
2954 014444 016100
2955 014446
2956 014446 104406
2957 014450 000745
2958 014452
2959 014452
2960 014452 104403
2961
2962 014454
2963 014454 104402
2964 014456 005037 002324
2965 014462
2966 014462 104443
2967 014464 000406
2968 014466 015756
2969 014470 000042
2970 014472 016241
2971 014474 177777
2972 014476 000001
2973 014500 000010
2974 014502
2975 014502 013737 015756 002302
2976 014510 012702 002350
2977 014514 004737 004324
2978 014520
2979 014520 104443
2980 014522 000404
2981 014524 002254
2982 014526 000130
2983 014530 016461
2984 014532 000001

1$: JMP EPRIN ;TEST ANY ADDITIONAL MEMORY
JMP P1 ;FIND ANY ADDITIONAL MEMORY
EXIT SUB ;END OF SUBTEST
TRAP C$EXIT
.WORD L10100-
E1: ERRDF 35,EXPND,CKSME
TRAP C$ERRDF
.WORD 35
.WORD EXPND
.WORD CKSME
CKLOOP
TRAP C$CLP1
BR MORE
E2: ERRDF 36,EXPND,CWKDE
TRAP C$ERRDF
.WORD 36
.WORD EXPND
.WORD CWKDE
CKLOOP
TRAP C$CLP1
BR MORE
E3: ERRDF 37,EXPND,NONXT
TRAP C$ERRDF
.WORD 37
.WORD EXPND
.WORD NONXT
CKLOOP
TRAP C$CLP1
BR MORE
L10100: ENDSUB
TRAP C$ESUB
EPRMT: BGNSUB
TRAP C$BSUB
CLR ERRFLG ;CLEAR ERROR FLAG
GMANID RWDCT,WORD,D,-1,1,10,NO
TRAP C$GMAN
BR 10000$
.WORD WORD
.WORD T$CODE
.WORD RWDCT
.WORD -1
.WORD T$LOLIM
.WORD T$HILIM
10000$: MOV WORD,WORDCT ;STORE CHECKWORD COUNT
MOV #EPROM,R2 ;POINTER TO STORAGE TABLE
JSR PC,INPUT ;INPUT CHECKWORDS
GMANIL EPADD,ANSR,1,YES
TRAP C$GMAN
BR 10001$
.WORD ANSR
.WORD T$CODE
.WORD EPADD
.WORD 1

```

2985	014534					10001\$:	TST	ANSR	: STANDARD MEMORY RANGE?
2986	014534	005737	002254				BNE	1\$: BR IF YES
2987	014540	001020					INC	ANSR	: RESTORE DEFAULT
2988	014542	005237	002254				GMANID	LOADR,STORE,D,-1,0,30,NO	
2989	014546						TRAP	C\$GMAN	
2990	014546	104443					BR	10002\$	
2991	014550	000406					.WORD	STORE	
2992	014552	002300					.WORD	T\$CODE	
2993	014554	000042					.WORD	LOADR	
2994	014556	003052					.WORD	-1	
2995	014560	177777					.WORD	T\$LOLIM	
2996	014562	000000					.WORD	T\$HILIM	
2997	014564	000030							
2998	014566					10002\$:	JSR	PC,SETADR	: GET FIRST PAGE ADDRESS
2999	014566	004737	005122				MOV	LOPAG,LOD2	: STORE LOW PAGE NO.
3000	014572	013737	002250	015750			BR	LD2	: SKIP NEXT INSTRUCTION
3001	014600	000403				1\$:	MOV	#020440,LOD2	: STANDARD PAGE = 40,41 4-6K RANGE
3002	014602	012737	020440	015750		LD2:	MOV	LOD2,PCR	: LOAD STARTING ADDRESS
3003	014610	013737	015750	177520			MOV	WORD,COUNTR	: PAGE COUNT
3004	014616	013737	015756	002252			CLR	RFLAG	: INDICATE EPROM
3005	014624	005037	002256				MOV	#EPROM,R3	: POINT TO CHECKWORDS
3006	014630	012703	002350			EPRTST:	MOV	(R3)+,CKWD	: GET CHECKWORD FOR THIS PAGE
3007	014634	012337	002306				JSR	PC,MEMTST	: TEST MEMORY
3008	014640	004737	004644				TST	REAL	: DOES THE MEMORY EXIST?
3009	014644	005737	002246				BEQ	E6	: BR IF NO
3010	014650	001455					TST	ERRFLG	: ANY OTHER ERRORS?
3011	014652	005737	002324				BEQ	NONE	: BR IF NO
3012	014656	001421					JSR	PC,VIRTAD	: GET ADDRESS OF ERROR
3013	014660	004737	004504				TST	BCF	: LOW BYTE PAGE?
3014	014664	005737	002244				BNE	HIADD	: BR IF NO
3015	014670	001004					MOV	#LOBYT,BYTLOC	: SET POINTER FOR ERROR MSG.
3016	014672	012737	002745	002322			BR	PRIOUT	: PRINT ERROR MESSAGE
3017	014700	000403				HIADD:	MOV	#HIBYT,BYTLOC	: POINTER FOR ERROR MSG.
3018	014702	012737	003007	002322		PRIOUT:	CMP	#1,ERRFLG	: CHECKSUM ERROR?
3019	014710	022737	000001	002324			BEQ	E4	: BR IF YES
3020	014716	001416					BR	E5	: ELSE CHECKWORD ERROR
3021	014720	000423				NONE:	ADD	#1002,PCR	: ADJUST PAGE IN PCR
3022	014722	062737	001002	177520			DEC	COUNTR	: DEC PAGE COUNT
3023	014730	005337	002252				BNE	EPRTST	: LOOP UNTIL FINISHED
3024	014734	001337				ADDTL:	TST	PASS	: FIRST PASS?
3025	014736	005737	002264				BNE	1\$: BR IF NO
3026	014742	001002					JMP	SYSRIN	: TEST ANY ADDITIONAL MEMORY
3027	014744	000137	014036			1\$:	JMP	P2	: FIND ANY ADDITIONAL MEMORY
3028	014750	000137	013732			E4:	ERRDF	40,EPRM,CKSME	
3029	014754						TRAP	C\$ERDF	
3030	014754	104455					.WORD	40	
3031	014756	000050					.WORD	EPRM	
3032	014760	016362					.WORD	CKSME	
3033	014762	015760					CKLOOP		
3034	014764						TRAP	C\$CLP1	
3035	014764	104406					BR	ADDTL	
3036	014766	000763				E5:	ERRDF	41,EPRM,CWKDE	
3037	014770						TRAP	C\$ERDF	
3038	014770	104455					.WORD	41	
3039	014772	000051					.WORD	EPRM	
3040	014774	016362							

3041	014776	016016				.WORD	CWKDE	
3042	015000					CKLOOP		
3043	015000	104406				TRAP	C\$CLP1	
3044	015002	000755				BR	ADDTL	
3045	015004		E6:			ERRDF	42,EPRM,NONXT	
3046	015004	104455				TRAP	C\$ERDF	
3047	015006	000052				.WORD	42	
3048	015010	016362				.WORD	EPRM	
3049	015012	016100				.WORD	NONXT	
3050	015014					CKLOOP		
3051	015014	104406				TRAP	C\$CLP1	
3052	015016	000747				BR	ADDTL	
3053	015020					ENDSUB		
3054	015020		L10101:					
3055	015020	104403				TRAP	C\$ESUB	
3056								
3057	015022					BGN SUB		
3058	015022	104402				TRAP	C\$BSUB	
3059	015024	005037	002324			CLR	ERRFLG	:CLEAR ERROR FLAG
3060	015030					GMANID	RMDCT,RESPND,D,-1,10,100,NO	
3061	015030	104443				TRAP	C\$GMAN	
3062	015032	000406				BR	000\$	
3063	015034	002312				.WORD		
3064	015036	000042				.WORD		
3065	015040	016241				.WORD		
3066	015042	177777				.WORD		
3067	015044	000010				.WORD		
3068	015046	000100				.WORD		
3069	015050							
3070	015050	013737	002312	015744		MOV		:STORE PAGE COUNT
3071	015056	013737	002312	002302		MOV		:COPY WORD COUNT
3072	015064	012702	002370			MOV		:POINTER TO STORAGE TABLE
3073	015070	004737	004324			JSR		:INPUT CHECKWORDS
3074	015074					GMANIL		
3075	015074	104443				TRAP		
3076	015076	000404				BR	10001\$	
3077	015100	002254				.WORD	ANSR	
3078	015102	000130				.WORD	T\$CODE	
3079	015104	016500				.WORD	SRR	
3080	015106	000001				.WORD	1	
3081	015110							
3082	015110	005737	002254			TST	ANSR	:STANDARD MEMORY RANGE?
3083	015114	001020				BNE	1\$:BR IF YES
3084	015116	005237	002254			INC	ANSR	:RESTORE DEFAULT VALUE
3085	015122					GMANID	LOADR,STORE,D,-1,0,30,NO	
3086	015122	104443				TRAP	C\$GMAN	
3087	015124	000406				BR	10002\$	
3088	015126	002300				.WORD	STORE	
3089	015130	000042				.WORD	T\$CODE	
3090	015132	003052				.WORD	LOADR	
3091	015134	177777				.WORD	-1	
3092	015136	000000				.WORD	T\$LOLIM	
3093	015140	000030				.WORD	T\$HILIM	
3094	015142							
3095	015142	004737	005122			JSR	PC,SETADR	:GET FIRST PAGE ADDRESS
3096	015146	013737	002250	015752		MOV	LOPAG,LOD3	:STORE LOW PAGE NO.

3097	015154	000403				BR	LD3	:SKIP NEXT INSTRUCTION
3098	015156	012737	100600	015752	1\$:	MOV	#100600,LOD3	:STANDARD PAGE = 200,201 16-32K RANGE
3099	015164	013737	015752	177520	LD3:	MOV	LOD3,PCR	:LOAD STARTING ADDRESS
3100	015172	012737	000001	002256		MOV	#1,RFLAG	:INDICATE ROM
3101	015200	012703	002370			MOV	#SYSROM,R3	:POINT TO CHECKWORDS
3102	015204	013737	015744	002252		MOV	PGCT,COUNTR	:PAGE COUNT
3103	015212	012337	002306		SYRTST:	MOV	(R3)+,CKWD	:GET CHECKWORD FOR THIS PAGE
3104	015216	004737	004644			JSR	PC,MEMTST	:TEST MEMORY
3105	015222	005737	002246			TST	REAL	:DOES THE MEMORY EXIST?
3106	015226	001450				BEQ	E11	:BR IF NO
3107	015230	005737	002324			TST	ERRFLG	:ANY OTHER ERRORS?
3108	015234	001421				BEQ	PASSED	:BR IF NO
3109	015236	004737	004504			JSR	PC,VIRTAD	:GET ADDRESS OF ERROR
3110	015242	005737	002244			TST	BCF	:LOW BYTE PAGE?
3111	015246	001004				BNE	HIGHB	:BR IF NO
3112	015250	012737	002745	002322		MOV	#LOBYT,BYTLOC	:SET POINTER FOR ERROR MSG.
3113	015256	000403				BR	MSGOUT	:PRINT ERROR MESSAGE
3114	015260	012737	003007	002322	HIGHB:	MOV	#HIBYT,BYTLOC	:POINTER FOR ERROR MSG.
3115	015266	022737	000001	002324	MSGOUT:	CMP	#1,ERRFLG	:CHECKSUM ERROR?
3116	015274	001411				BEQ	E7	:BR IF YES
3117	015276	000416				BR	E10	:ELSE CHECKWORD ERROR
3118	015300	062737	001002	177520	PASSED:	ADD	#1002,PCR	:ADJUST PAGE IN PCR
3119	015306	005337	002252			DEC	COUNTR	:DEC PAGE COUNT
3120	015312	001337				BNE	SYRTST	:LOOP UNTIL FINISHED
3121	015314				NEXT:	EXIT	TST	:TEST IS FINISHED
3122	015314	104432				TRAP	C\$EXIT	
3123	015316	001244				.WORD	L10076-	
3124	015320				E7:	ERRDF	43,SYSR,CKSME	
3125	015320	104455				TRAP	C\$ERDF	
3126	015322	000053				.WORD	43	
3127	015324	016403				.WORD	SYSR	
3128	015326	015760				.WORD	CKSME	
3129	015330					CKLOOP		
3130	015330	104406				TRAP	C\$CLP1	
3131	015332	000770				BR	NEXT	
3132	015334				E10:	ERRDF	44,SYSR,CWKDE	
3133	015334	104455				TRAP	C\$ERDF	
3134	015336	000054				.WORD	44	
3135	015340	016403				.WORD	SYSR	
3136	015342	016016				.WORD	CWKDE	
3137	015344					CKLOOP		
3138	015344	104406				TRAP	C\$CLP1	
3139	015346	000762				BR	NEXT	
3140	015350				E11:	ERRDF	45,SYSR,NONXT	
3141	015350	104455				TRAP	C\$ERDF	
3142	015352	000055				.WORD	45	
3143	015354	016403				.WORD	SYSR	
3144	015356	016100				.WORD	NONXT	
3145	015360					CKLOOP		
3146	015360	104406				TRAP	C\$CLP1	
3147	015362	000754				BR	NEXT	
3148	015364					ENDSUB		
3149	015364				L10102:			
3150	015364	104403				TRAP	C\$ESUB	
3151								
3152	015366					BGNSUB		

3153	015366	104402				TRAP	C\$BSUB	
3154	015370	005037	002324			SYSET: CLR	ERRFLG	;CLEAR ERROR FLAG
3155	015374					GMANID	RWDCT,RESPND,D,-1,10,40,NO	
3156	015374	104443				TRAP	C\$GMAN	
3157	015376	000406				BR	10000\$	
3158	015400	002312				.WORD	RESPND	
3159	015402	000042				.WORD	T\$CODE	
3160	015404	016241				.WORD	RWDCT	
3161	015406	177777				.WORD	-1	
3162	015410	000010				.WORD	T\$LOLIM	
3163	015412	000040				.WORD	T\$HILIM	
3164	015414					10000\$:		
3165	015414	013737	002312	015744		MOV	RESPND,PGCT	;STORE PAGE COUNT
3166	015422	013737	002312	002302		MOV	RESPND,WORDCT	;COPY WORD COUNT
3167	015430	012702	002370			MOV	#SYSROM,R2	;POINTER TO STORAGE TABLE
3168	015434	004737	004324			JSR	PC,INPUT	;INPUT CHECKWORDS
3169	015440					GMANIL	SYEE,ANSR,1,YES	
3170	015440	104443				TRAP	C\$GMAN	
3171	015442	000404				BR	10001\$	
3172	015444	002254				.WORD	ANSR	
3173	015446	000130				.WORD	T\$CODE	
3174	015450	016530				.WORD	SYEE	
3175	015452	000001				.WORD	1	
3176	015454					10001\$:		
3177	015454	005737	002254			TST	ANSR	;STANDARD MEMORY RANGE?
3178	015460	001020				BNE	1\$;BR IF YES
3179	015462	005237	002254			INC	ANSR	;RESTORE DEFAULT VALUE
3180	015466					GMANID	LOADR,STORE,D,-1,0,30,NO	
3181	015466	104443				TRAP	C\$GMAN	
3182	015470	000406				BR	10002\$	
3183	015472	002300				.WORD	STORE	
3184	015474	000042				.WORD	T\$CODE	
3185	015476	003052				.WORD	LOADR	
3186	015500	177777				.WORD	-1	
3187	015502	000000				.WORD	T\$LOLIM	
3188	015504	000030				.WORD	T\$HILIM	
3189	015506					10002\$:		
3190	015506	004737	005122			JSR	PC,SETADR	;GET FIRST PAGE ADDRESS
3191	015512	013737	002250	015754		MOV	LOPAG,LOD4	;STORE LOW PAGE NO.
3192	015520	000403				BR	LD4	;SKIP NEXT INSTRUCTION
3193	015522	012737	100600	015754		1\$:	MOV #100600,LOD4	;STANDARD PAGE = 200,201 16-24K RANGE
3194	015530	013737	015754	177520		LD4:	MOV LOD4,PCR	;LOAD STARTING ADDRESS
3195	015536	005037	002256			CLR	RFLAG	;INDICATE EPROM
3196	015542	012703	002370			MOV	#SYSROM,R3	;POINT TO CHECKWORDS
3197	015546	013737	015744	002252		MOV	PGCT,COUNTR	;PAGE COUNT
3198	015554	012337	002306			SYETST: MOV	(R3)+,CKWD	;GET CHECKWORD FOR THIS PAGE
3199	015560	004737	004644			JSR	PC,MEMTST	;TEST MEMORY
3200	015564	005737	002246			TST	REAL	;DOES THIS MEMORY EXIST?
3201	015570	001452				BEQ	E14	;BR IF NO
3202	015572	005737	002324			TST	ERRFLG	;ANY ERRORS?
3203	015576	001421				BEQ	CONT	;BR IF NO
3204	015600	004737	004504			JSR	PC,VIRTAD	;GET ADDRESS OF ERROR
3205	015604	005737	002244			TST	BCF	;LOW BYTE PAGE?
3206	015610	001004				BNE	HBYTE	;BR IF NO
3207	015612	012737	002745	002322		MOV	#LOBYT,BYTLOC	;SET POINTER FOR ERROR MSG.
3208	015620	000403				BR	PRIN	;PRINT ERROR MESSAGE

```

3209 015622 012737 003007 002322 HBYTE: MOV #HIBYT,BYTLOC ;POINTER FOR ERROR MSG.
3210 015630 022737 000001 002324 PRIN: CMP #1,ERRFLG ;CHECKSUM ERROR?
3211 015636 001411 BEQ E12 ;BR IF YES
3212 015640 000417 BR E13 ;ELSE CHECKWORD ERROR
3213 015642 062737 001002 177520 CONT: ADD #1002,PCR ;ADJUST PAGE IN PCR
3214 015650 005337 002252 DEC COUNTR ;DEC PAGE COUNT
3215 015654 001337 BNE SYETST ;LOOP UNTIL FINISHED
3216 015656 EXIT TST ;TEST IS FINISHED
3217 015656 104432 TRAP C$EXIT
3218 015660 000702 .WORD L10076-.
3219 015662 E12: ERRDF 46,SYSE,CKSME
3220 015662 104455 TRAP C$ERDF
3221 015664 000056 .WORD 46
3222 015666 016416 .WORD SYSE
3223 015670 015760 .WORD CKSME
3224 015672 CKLOOP
3225 015672 104406 TRAP C$CLP1
3226 015674 EXIT TST
3227 015674 104432 TRAP C$EXIT
3228 015676 000664 .WORD L10076-.
3229 015700 E13: ERRDF 47,SYSE,CWKDE
3230 015700 104455 TRAP C$ERDF
3231 015702 000057 .WORD 47
3232 015704 016416 .WORD SYSE
3233 015706 016016 .WORD CWKDE
3234 015710 CKLOOP
3235 015710 104406 TRAP C$CLP1
3236 015712 EXIT TST
3237 015712 104432 TRAP C$EXIT
3238 015714 000646 .WORD L10076-.
3239 015716 E14: ERRDF 50,SYSE,NONXT
3240 015716 104455 TRAP C$ERDF
3241 015720 000062 .WORD 50
3242 015722 016416 .WORD SYSE
3243 015724 016100 .WORD NONXT
3244 015726 CKLOOP
3245 015726 104406 TRAP C$CLP1
3246 015730 EXIT TST
3247 015730 104432 TRAP C$EXIT
3248 015732 000630 .WORD L10076-.
3249 015734 ENDSUB
3250 015734 L10103:
3251 015734 104403 TRAP C$ESUB
3252 015736 EXIT TST
3253 015736 104432 TRAP C$EXIT
3254 015736 000622 .WORD L10076-.
3255 015740
3256 015742 000000 ADDON: .WORD 0
3257 015744 000000 PGCT: .WORD 0
3258 015746 000000 LOD1: .WORD 0
3259 015750 000000 LOD2: .WORD 0
3260 015752 000000 LOD3: .WORD 0
3261 015754 000000 LOD4: .WORD 0
3262 015756 000000 WORD: .WORD 0
3263
3264

```

3265					
3266	015760				BGNMSG CKSME
3267	015760				CKSME::
3268	015760				PRINTB #ERM6,BYTLOC
3269	015760	013746	002322		MOV BYTLOC,-(SP)
3270	015764	012746	016162		MOV #ERM6,-(SP)
3271	015770	012746	000002		MOV #2,-(SP)
3272	015774	010600			MOV SP,R0
3273	015776	104414			TRAP C\$PNTB
3274	016000	062706	000006		ADD #6,SP
3275	016004	004737	004504		JSR PC,VIRTAD
3276	016010	004737	004230		JSR PC,VIPRI
3277	016014				ENDMSG
3278	016014				L10104:
3279	016014	104423			TRAP C\$MSG
3280					
3281	016016				BGNMSG CWKDE
3282	016016				CWKDE::
3283	016016				PRINTB #ERMS
3284	016016	012746	016132		MOV #ERMS,-(SP)
3285	016022	012746	000001		MOV #1,-(SP)
3286	016026	010600			MOV SP,R0
3287	016030	104414			TRAP C\$PNTB
3288	016032	062706	000004		ADD #4,SP
3289	016036	004737	004504		JSR PC,VIRTAD
3290	016042	004737	004230		JSR PC,VIPRI
3291	016046				PRINTB #REGDT,CKWD,BADWD
3292	016046	013746	002310		MOV BADWD,-(SP)
3293	016052	013746	002306		MOV CKWD,-(SP)
3294	016056	012746	003646		MOV #REGDT,-(SP)
3295	016062	012746	000003		MOV #3,-(SP)
3296	016066	010600			MOV SP,R0
3297	016070	104414			TRAP C\$PNTB
3298	016072	062706	000010		ADD #10,SP
3299	016076				ENDMSG
3300	016076				L10105:
3301	016076	104423			TRAP C\$MSG
3302					
3303	016100				BGNMSG NONXT
3304	016100				NONXT::
3305	016100				PRINTB #LOST
3306	016100	012746	016211		MOV #LOST,-(SP)
3307	016104	012746	000001		MOV #1,-(SP)
3308	016110	010600			MOV SP,R0
3309	016112	104414			TRAP C\$PNTB
3310	016114	062706	000004		ADD #4,SP
3311	016120	004737	004504		JSR PC,VIRTAD
3312	016124	004737	004230		JSR PC,VIPRI
3313	016130				ENDMSG
3314	016130				L10106:
3315	016130	104423			TRAP C\$MSG
3316					
3317	016132	040445	047111	047503	ERM5: .ASCIZ /%AINCORRECT CHECKWORD%N/
3318	016140	051122	041505	020124	
3319	016146	044103	041505	053513	
3320	016154	051117	022504	000116	

HARDWARE TESTS MACY11 30(1046)
 CVM8AD.P11 10-JUL-81 17:08

20-JUL-81 11:44 PAGE 68
 TEST 7: TEST ALL ADDITIONAL MEMORY

SEQ 0068

3321					
3322	016162	040445	044103	041505	ERM6: .ASCIZ /%ACHECKSUM ERROR%N%T%N/
3323	016170	051513	046525	042440	
3324	016176	051122	051117	047045	
3325	016204	052045	047045	000	
3326					
3327	016211	045	047101	047117	LOST: .ASCIZ /%ANON-EXISTENT MEMORY%N/
3328	016216	042455	044530	052123	
3329	016224	047105	020124	042515	
3330	016232	047515	054522	047045	
3331	016240	000			
3332					
3333	016241	110	053517	046440	RWDCT: .ASCIZ /HOW MANY CHECKWORDS WILL BE INPUT/
3334	016246	047101	020131	044103	
3335	016254	041505	053513	051117	
3336	016262	051504	053440	046111	
3337	016270	020114	042502	044440	
3338	016276	050116	052125	000	
3339					
3340	016303	101	054516	040440	EXEC: .ASCIZ /ANY ADDITIONAL MEMORY /
3341	016310	042104	052111	047511	
3342	016316	040516	020114	042515	
3343	016324	047515	054522	000040	
3344					
3345	016332	054105	040520	042116	FXPND: .ASCIZ /EXPANDED DIAGNOSTIC ROM/
3346	016340	042105	042040	040511	
3347	016346	047107	051517	044524	
3348	016354	020103	047522	000115	
3349					
3350	016362	050105	047522	020115	EPRM: .ASCIZ /EPROM IN SOCKETS/
3351	016370	047111	051440	041517	
3352	016376	042513	051524	000	
3353					
3354	016403	123	051531	042524	SYSR: .ASCIZ /SYSTEM ROM/
3355	016410	020115	047522	000115	
3356					
3357	016416	054523	052123	046505	SYSE: .ASCIZ /SYSTEM EPROM/
3358	016424	042440	051120	046517	
3359	016432	000			
3360					
3361	016433	105	050130	047101	EXADD: .ASCIZ /EXPANDED ROM IN 2-4K /
3362	016440	042504	020104	047522	
3363	016446	020115	047111	031040	
3364	016454	032055	020113	000	
3365					
3366	016461	105	051120	046517	EPADD: .ASCIZ /EPROM IN 4-6K /
3367	016466	044440	020116	026464	
3368	016474	045466	000040		
3369					
3370	016500	054523	052123	046505	SRR: .ASCIZ /SYSTEM ROM START AT 16K/
3371	016506	051040	046517	051440	
3372	016514	040524	052122	040440	
3373	016522	020124	033061	000113	
3374					
3375	016530	054523	052123	046505	SYEE: .ASCIZ /SYSTEM EPROM START AT 16K/
3376	016536	042440	051120	046517	

3377 016544 051440 040524 052122
3378 016552 040440 020124 033061
3379 016560 000113
3380
3381
3382
3383 016562
3384 016562
3385 016562 104401
3386
3387
3388
3389
3390
3391
3392
3393
3394

L10076: .EVEN
ENDTST
TRAP C\$ETST

PARAMETER CODING MACY11 30(1046) 20-JUL-81 11:44 PAGE 70 F 6
CVM8AD.P11 10-JUL-81 17:08 TEST 7: TEST ALL ADDITIONAL MEMORY

SEQ 0070

3395
3396
3397
3398

.TITLE PARAMETER CODING
.SBTTL IDENTIFICATION

.SBTTL HARDWARE PARAMETER CODING SECTION

;++
: THE HARDWARE PARAMETER CODING SECTION CONTAINS MACROS
: THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES. THE
: MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
: INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES. THE
: MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
: WITH THE OPERATOR.
:--

3399
3400
3401
3402
3403
3404
3405
3406
3407
3408
3409
3410
3411 016564
3412 016564 000107
3413 016566
3414
3415 016566
3416 016566 000032
3417 016570 016646
3418 016572 160000
3419 016574 000000
3420 016576 000016
3421 016600
3422 016600 001032
3423 016602 016662
3424 016604 177777
3425 016606 000066
3426 016610 000100
3427 016612
3428 016612 002032
3429 016614 016713
3430 016616 177777
3431 016620 000006
3432 016622 000007
3433 016624
3434 016624 003032
3435 016626 016733
3436 016630 177777
3437 016632 000000
3438 016634 007777
3439 016636
3440 016636 004120
3441 016640 016762
3442 016642 000001
3443
3444 016644
3445 016644 060004
3446
3447 016646 047125 052111 047040
3448 016654 046525 042502 000122
3449 016662 047111 042524 051122
3450 016670 050125 020124 042526
3451 016676 052103 051117 040440
3452 016704 042104 042522 051523
3453 016712 000
3454 016713 111 052116 051105

BGNHRD
.WORD L10107-L\$HARD/2
L\$HARD::
GPRMD UNIT,0,0,160000,0,16,YES
.WORD T\$CODE
.WORD UNIT
.WORD 160000
.WORD T\$LOLIM
.WORD T\$HILIM
GPRMD INTVEC,2,0,-1,66,100,YES
.WORD T\$CODE
.WORD INTVEC
.WORD -1
.WORD T\$LOLIM
.WORD T\$HILIM
GPRMD LEV,4,0,-1,6,7,YES
.WORD T\$CODE
.WORD LEV
.WORD -1
.WORD T\$LOLIM
.WORD T\$HILIM
GPRMD RKSW,6,0,-1,0,7777,YES
.WORD T\$CODE
.WORD RKSW
.WORD -1
.WORD T\$LOLIM
.WORD T\$HILIM
GPRML PAX,10,1,NO
.WORD T\$CODE
.WORD PAX
.WORD 1
EXIT HRD
.WORD T\$CODE
UNIT: .ASCIZ /UNIT NUMBER/
INTVEC: .ASCIZ /INTERRUPT VECTOR ADDRESS/
LEV: .ASCIZ /INTERRUPT LEVEL/

PARAMETER CODING
CVMBAD.P11

10-JUL-81

MACY11 30(1046)
17:08

20-JUL-81 11:44 PAGE 72
H 6
HARDWARE PARAMETER CODING SECTION

SEQ 0072

3455	016720	052522	052120	046040
3456	016726	053105	046105	000
3457	016733	122	041517	042513
3458	016740	020122	053523	052111
3459	016746	044103	051440	052105
3460	016754	044524	043516	000123
3461	016762	042524	052123	047111
3462	016770	020107	020101	042113
3463	016776	030506	026461	000102
3464				
3465				
3466	017004			
3467				
3468	017004			

RKSW: .ASCIZ /ROCKER SWITCH SETTINGS/

PAX: .ASCIZ /TESTING A KDF11-B/

.EVEN

ENDHRD

.EVEN

L10107:

.SBTTL SOFTWARE PARAMETER CODING SECTION

++
: THE SOFTWARE PARAMETER CODING SECTION CONTAINS MACROS
: THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES. THE
: MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
: INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES. THE
: MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
: WITH THE OPERATOR.
:--

3469
3470
3471
3472
3473
3474
3475
3476
3477
3478
3479
3480 017004
3481 017004 000161
3482 017006
3483 017006
3484 017006 000032
3485 017010 017130
3486 017012 177777
3487 017014 000000
3488 017016 177777
3489 017020
3490 017020 001032
3491 017022 017205
3492 017024 177777
3493 017026 000000
3494 017030 177777
3495 017032
3496 017032 002032
3497 017034 017223
3498 017036 177777
3499 017040 000000
3500 017042 177777
3501 017044
3502 017044 003032
3503 017046 017241
3504 017050 177777
3505 017052 000000
3506 017054 177777
3507 017056
3508 017056 004032
3509 017060 017257
3510 017062 177777
3511 017064 000000
3512 017066 177777
3513 017070
3514 017070 005032
3515 017072 017275
3516 017074 177777
3517 017076 000000
3518 017100 177777
3519 017102
3520 017102 006032
3521 017104 017313
3522 017106 177777
3523 017110 000000
3524 017112 177777

BGNSFT
.WORD L10110-L\$SOFT/2
L\$SOFT: :
GPRMD CKW1,0,0,-1,0,177777,YES
.WORD T\$CODE
.WORD CKW1
.WORD -1
.WORD T\$LOLIM
.WORD T\$HILIM
GPRMD CKW2,2,0,-1,0,177777,YES
.WORD T\$CODE
.WORD CKW2
.WORD -1
.WORD T\$LOLIM
.WORD T\$HILIM
GPRMD CKW3,4,0,-1,0,177777,YES
.WORD T\$CODE
.WORD CKW3
.WORD -1
.WORD T\$LOLIM
.WORD T\$HILIM
GPRMD CKW4,6,0,-1,0,177777,YES
.WORD T\$CODE
.WORD CKW4
.WORD -1
.WORD T\$LOLIM
.WORD T\$HILIM
GPRMD CKW5,10,0,-1,0,177777,YES
.WORD T\$CODE
.WORD CKW5
.WORD -1
.WORD T\$LOLIM
.WORD T\$HILIM
GPRMD CKW6,12,0,-1,0,177777,YES
.WORD T\$CODE
.WORD CKW6
.WORD -1
.WORD T\$LOLIM
.WORD T\$HILIM
GPRMD CKW7,14,0,-1,0,177777,YES
.WORD T\$CODE
.WORD CKW7
.WORD -1
.WORD T\$LOLIM
.WORD T\$HILIM

```

3525 017114          GPRMD  CKW8,16,0,-1,0,177777,YES
3526 017114 007032  .WORD  T$CODE
3527 017116 017331  .WORD  CKW8
3528 017120 177777  .WORD  -1
3529 017122 000000  .WORD  T$LOLIM
3530 017124 177777  .WORD  T$HILIM
3531
3532 017126          EXIT SFT
3533 017126 111004  .WORD  T$CODE
3534
3535 017130 044103 041505 053513 CKW1:  .ASCIZ  /CHECKWORDS FOR DIAGNOSTIC ROM. CHECKWORD 1: /
3536 017136 051117 051504 043040
3537 017144 051117 042040 040511
3538 017152 047107 051517 044524
3539 017160 020103 047522 027115
3540 017166 041440 042510 045503
3541 017174 047527 042122 030440
3542 017202 020072 000
3543 017205 103 042510 045503 CKW2:  .ASCIZ  /CHECKWORD 2: /
3544 017212 047527 042122 031040
3545 017220 020072 000
3546 017223 103 042510 045503 CKW3:  .ASCIZ  /CHECKWORD 3: /
3547 017230 047527 042122 031440
3548 017236 020072 000
3549 017241 103 042510 045503 CKW4:  .ASCIZ  /CHECKWORD 4: /
3550 017246 047527 042122 032040
3551 017254 020072 000
3552 017257 103 042510 045503 CKW5:  .ASCIZ  /CHECKWORD 5: /
3553 017264 047527 042122 032440
3554 017272 020072 000
3555 017275 103 042510 045503 CKW6:  .ASCIZ  /CHECKWORD 6: /
3556 017302 047527 042122 033040
3557 017310 020072 000
3558 017313 103 042510 045503 CKW7:  .ASCIZ  /CHECKWORD 7: /
3559 017320 047527 042122 033440
3560 017326 020072 000
3561 017331 103 042510 045503 CKW8:  .ASCIZ  /CHECKWORD 8: /
3562 017336 047527 042122 034040
3563 017344 020072 000
3564          017350          .EVEN
3565
3566
3567 017350          ENDSFT
3568          .EVEN
3569 017350          L10110:
3570
3571          017420          .=.+50
3572 017420
3573          LASTAD
3574 017420 000000          .EVEN
3575 017422 000000          .WORD  0
3576 017424          .WORD  0
3577          L$LAST::
3578          000001          .END

```


PARAMETER CODING
CVM8AD.P11

10-JUL-81

MACY11 30(1046)
17:08

20-1
CRO^s

M 6
11:44 PAGE 78
REFERENCE TABLE -- USER SYMBOLS

SEQ 0077

C\$GPRI= 000040	380#																
C\$INIT= 000011	380#	1288															
C\$INLP= 000020	380#																
C\$MANI= 000050	380#	2184	2231	2381	2550	2796											
C\$MEM = 000031	380#																
C\$MSG = 000023	380#	892	912	932	952	964	976	988	2041	2060	2079	2098	2117				
	2339	2600	2613	2626	2639	2652	3279	3301	3315								
C\$OPEN= 000034	380#																
C\$PNTB= 000014	380#	880	900	920	940	960	972	984	2030	2037	2049	2056	2068				
	2075	2087	2094	2106	2113	2335	2596	2609	2622	2635	2647	3273	3287				
	3297	3309															
C\$PNTF= 000017	380#	998	1047	2278	2290	2306	2316	2525	2534	2543	2564	2573					
C\$PNTS= 000016	380#																
C\$PNTX= 000015	380#	888	908	928	948												
C\$QIO = 000377	380#																
C\$RDBU= 000007	380#																
C\$REFG= 000047	380#																
C\$RESE= 000033	380#	1988															
C\$REVI= 000003	380#	445															
C\$RFLA= 000021	380#																
C\$RPT = 000025	380#	1261															
C\$SEFG= 000046	380#																
C\$SPRI= 000041	380#	1283	1875	1880	1900	1905	1933	1938	1958	1963	1991	1996					
C\$SVEC= 000037	380#	1870															
C\$TPRI= 000013	380#																
DADDR 013622	2391	2754#															
DATOUT 014336	2919	2921#															
DEL CNT 002326	704#	1225*	1227														
DERR1 012474	G 2422	2590#															
DERR2 012522	G 2439	2603#															
DERR3 012550	G 2456	2616#															
DERR4 012576	G 2473	2629#															
DERR5 012624	G 2557	2642#															
DFLTST 014112	2798	2811	2822	2864	2866#												
DFPTBL 002144	G 520#																
DIAGER 003534	841#	981															
DIAGMC= 000000	380																
DIAIN 013766	2823#																
DLOOP 011676	2415#	2485															
DRLP 012472	2396*	2409*	2410	2587#													
DRTST 011656	2412#																
DSPCOD 002122	G 496#																
EF.CON= 000036	G 632#																
EF.NEW= 000035	G 633#																
EF.PWR= 000034	G 634#																
EF.RES= 000037	G 631#																
EF.STA= 000040	G 630#																
EPADD 016461	2983	3366#															
EPRIN 014010	2833#	2929															
EPRM 016362	2838	3032	3040	3048	3350#												
EPRMT 014456	2843	2964#															
EPROM 002350	706#	2976	3006														
EPRTST 014634	3007#	3024															
ERM5 016132	3284	3317#															
ERM6 016162	3270	3322#															
ERRFLG 002324	703#	1138*	1155*	1166*	1171*	2876*	2913	2921	2964*	3011	3019	3059*	3107				

EVL = 000004 G	3115	3154*	3202	3210											
EXADD 016433	650#														
EXEC 016303	2885	3361#													
EXPDIA 002330	2818	3340#													
EXPND 016332	705#	2878	2907												
EXFROM 014122	2828	2937	2945	2953	3345#										
EXPSUM 002260	2832	2876#													
EXPTST 014262	685#	1134*	1135*	1136	1151*	1152*	1153	2431*	2432*	2433	2465*	2466*	2467		
E\$END = 002100	2909#	2926													
E\$LOAD= 000035	380#														
E1 014406	380#	469													
E10 015334	2922	2934#													
E11 015350	3117	3132#													
E12 015662	3106	3140#													
E13 015700	3211	3219#													
E14 015716	3212	3229#													
E2 014422	3201	3239#													
E3 014436	2923	2942#													
E4 014754	2912	2950#													
E5 014770	3020	3029#													
E6 015004	3021	3037#													
E7 015320	3010	3045#													
F\$AU = 000015	3116	3124#													
F\$AUTO= 000020	380#														
F\$BGN = 000040	380#	1298	1300												
	955	390	487	496	508	594	668	676	709	875	895	915	935		
	1371	967	979	1258	1272	1291	1298	1310	1325	1347	1349	1362	1367		
	1475	1385	1390	1394	1408	1413	1417	1431	1436	1440	1452	1457	1461		
	1582	1480	1484	1498	1503	1507	1521	1526	1530	1548	1553	1557	1577		
	1672	1586	1595	1599	1603	1616	1621	1625	1639	1644	1649	1663	1668		
	1776	1686	1691	1695	1709	1714	1717	1731	1736	1739	1753	1758	1762		
	1896	1781	1785	1803	1808	1812	1832	1837	1840	1854	1860	1864	1892		
	2044	1917	1925	1929	1950	1954	1975	1981	1985	2008	2011	2016	2025		
	2366	2063	2082	2101	2165	2174	2190	2212	2216	2229	2249	2320	2328		
	2578	2377	2379	2426	2443	2448	2460	2477	2481	2487	2493	2503	2509		
	2875	2584	2590	2603	2616	2629	2642	2760	2787	2789	2793	2867	2870		
	3250	2932	2959	2963	3054	3058	3122	3149	3153	3217	3227	3237	3247		
	380#	3254	3267	3282	3304	3384	3412	3445	3481	3533					
F\$CLEA= 000007	380#	1310	1331												
F\$DU = 000016	380#														
F\$END = 000041	380#	390	487	496	508	594	668	676	709	893	913	933	953		
	965	977	989	1262	1289	1302	1325	1333	1347	1349	1362	1367	1369		
	1371	1385	1390	1392	1394	1408	1413	1415	1417	1431	1436	1438	1440		
	1452	1457	1459	1461	1475	1480	1482	1484	1498	1503	1505	1507	1521		
	1526	1528	1530	1548	1553	1555	1557	1577	1582	1584	1586	1588	1595		
	1599	1603	1616	1621	1623	1625	1639	1644	1646	1649	1663	1668	1670		
	1672	1686	1691	1693	1695	1709	1714	1716	1717	1731	1736	1738	1739		
	1753	1758	1760	1762	1776	1781	1783	1785	1803	1808	1810	1812	1832		
	1837	1839	1840	1842	1854	1860	1864	1892	1894	1896	1917	1919	1925		
	1929	1950	1952	1954	1975	1977	1981	1985	2008	2010	2011	2021	2042		
	2061	2080	2099	2118	2165	2167	2174	2190	2212	2216	2218	2229	2249		
	2320	2340	2366	2368	2377	2379	2426	2443	2445	2448	2460	2477	2479		
	2481	2487	2489	2493	2503	2505	2509	2578	2580	2584	2601	2614	2627		
	2640	2653	2760	2762	2787	2789	2793	2867	2870	2872	2875	2932	2959		
	2961	2963	3054	3056	3058	3122	3149	3151	3153	3217	3227	3237	3247		
	3250	3252	3254	3280	3302	3316	3384	3386	3445	3469	3533	3570			

PARAMETER CODING
CVMBAD.P11

10-JUL-81

MACY11 30(1046)
17:08

20-JUL-81 11:44 PAGE 81
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0080

GSYES = 000010	380#	2390	2817	2884	2982	3078	3173	3416	3422	3428	3434	3484	3490
	3496	3502	3508	3514	3520	3526							
HBYTE 015622	3206	3209#											
HELP = 000000	6#	15	18	23	24	90	93	110	117	126	130	148	228
	236	274	279	298	379	385#	399	488	508	522	544	584#	665
	790	864	1003	1235	1237	1239	1241	1243	1245	1247	1249	1250	1251
	1253#	1284	1323	1328	1336#	3386	3388	3389	3391	3393	3395	3397#	3415
	3447	3534	3565										
HIADD 014702	3015	3018#											
HIBYT 003007	754#	2920	3018	3114	3209								
HIBYTE 004726	1131	1142#											
HIGH 014330	2917	2920#											
HIGHB 015260	3111	3114#											
HIRANG 002320	701#	993	1112*	1113*	1116*								
HIROM 013073	2632	2685#											
HOE = 100000 G	663#												
HRAER# 013014	2619	2676#											
IBE = 010000 G	660#												
ICOUNT 007650	1862*	1881	1890*	1906	1915*	1939	1948*	1964	1973*	1997	2006*	2017*	2022#
	2033	2052	2071	2090	2109								
IDU = 000040 G	653#												
IER = 020000 G	661#												
INLP 004344	1049#	1062											
INPUT 004324	1043#	2879	2977	3073	3168								
INSTR 004402	1044	1065#											
INTCT 010224	2034	2053	2072	2091	2110	2127#							
INTSR 007642	1867	2014#											
INTVEC 016662	3423	3449#											
INWORD 004470	1054	1075#											
ISR = 000100 G	654#												
IXE = 004000 G	659#												
I\$AU = 000041	380#												
I\$AUTO= 000041	380#	1298#	1302#										
I\$CLN = 000041	380#	1310#	1325	1333#									
I\$DU = 000041	380#												
I\$HRD = 000041	3412#	3469#											
I\$INIT= 000041	380#	1272#	1289#										
I\$MOD = 000041	380#	390#	487#	496#	508#	594#	668#	676#	709#				
I\$MSG = 000041	380#	875#	893#	895#	913#	915#	933#	935#	953#	955#	965#	967#	977#
	979#	989#	2025#	2042#	2044#	2061#	2063#	2080#	2082#	2099#	2101#	2118#	2328#
	2340#	2590#	2601#	2603#	2614#	2616#	2627#	2629#	2640#	2642#	2653#	3267#	3280#
	3282#	3302#	3304#	3316#									
I\$PROT= 000040	380#	1291#											
I\$PTAB= 000041	380#												
I\$PWR = 000041	380#												
I\$RPT = 000041	380#	1258#	1262#										
I\$SEG = 000041	380#	1347	1349	1371	1394	1417	1440	1461	1484	1507	1530	1557	1595
	1603	1625	1649	1672	1695	1717	1739	1762	1785	1812	1854	1864	1896
	1929	1954	1985	2174	2229	2377	2379	2448	2481	2493	2509	2787	2789
	2875	2963	3058	3153									
I\$SETU= 000041	380#												
I\$SFT = 000041	3481#	3570#											
I\$SRV = 000041	380#	2016#	2021#										
I\$SUB = 000041	380#	1347	1349#	1367#	1369#	1371#	1390#	1392#	1394#	1413#	1415#	1417#	1436#
	1438#	1440#	1457#	1459#	1461#	1480#	1482#	1484#	1503#	1505#	1507#	1526#	1528#
	1530#	1553#	1555#	1557#	1582#	1584#	1595	1603#	1621#	1623#	1625#	1644#	1646#

RERR1	003712	G	875#	1358	1612														
RERR2	003760	G	895#	1381	1635														
RERR3	004026	G	915#	1404	1471	1659	1727												
RERR4	004074	G	935#	1427	1448	1494	1517	1682	1705	1749	1772								
RERR5	004142	G	955#	1544	1799														
RERR6	004164	G	967#	1573	1828														
RESPND	002312		698#	1315*	2826	2831	2836	2841	2847	2852	2858	2863	3063	3070	3071				
			3158	3165	3166														
RFLAG	002256		684#	1114	2414*	2906*	3005*	3100*	3195*										
RKSW	016733		3435	3457#															
ROTL P1	006120		1534#	1539															
ROTL P2	006204		1561#	1567															
ROTL P3	007006		1789#	1794															
ROTL P4	007072		1816#	1822															
ROTO	003462		833#	969															
ROT1	003412		825#	957															
RSET	002314		699#	1314*	2389	2394													
RSTRT	011642		2385	2397	2410#														
RWDCT	016241		2970	3065	3160	3333#													
RWR	0C2570		723#	1357	1380	1403	1426	1447	1470	1493	1516	1543	1572						
RWREG =	177522		1313*	1344#	1350*	1353	1372*	1373	1376	1395*	1396	1399	1418*	1419	1422				
			1441*	1442	1462*	1463	1466	1485*	1486	1489	1508*	1509	1512	1531*	1532*				
			1533	1534	1538*	1558*	1559*	1560	1561	1564*									
SERR1	011424		2332	2351#															
SETADR	005122		1186#	2408	2901	2999	3095	3190											
SFPTBL	002160	G	542#	2413	2511														
SRR	016500		3079	3370#															
STORE	002300		693#	1052	1059	1186	2401	2894	2992	3088	3183								
STRT	011600		2383	2396#															
SVCGBL =	000000		380#	381#	390	391	401	402	410	411	412	413	414	415	416				
			417	418	419	420	421	422	423	424	425	426	427	428	429				
			430	431	432	433	434	435	436	437	438	439	440	441	442				
			443	444	445	447	448	450	451	452	453	454	455	456	457				
			458	459	460	461	462	463	464	465	466	467	468	469	470				
			471	472	473	474	475	476	477	478	479	480	481	482	483				
			484	485	496	497	499	500	519	520	521	541	542	543	594				
			595	676	677	774	775	784	785	875	876	895	896	915	916				
			935	936	955	956	967	968	979	980	1258	1259	1272	1273	1291				
			1292	1298	1299	1310	1311	2016	2017	2025	2026	2044	2045	2063	2064				
			2082	2083	2101	2102	2328	2329	2590	2591	2603	2604	2616	2617	2629				
			2630	2642	2643	3267	3268	3282	3283	3304	3305	3413	3414	3482	3483				
			3576#	3577															
SVCINS =	000000		380#	402	403	404	405	406	407	408	409	410	411	412	413				
			414	415	416	417	418	419	420	421	422	423	424	425	426				
			427	428	429	430	431	432	433	434	435	436	437	438	439				
			440	441	442	443	444	445	446	447	448	449	450	451	452				
			453	454	455	456	457	458	459	460	461	462	463	464	465				
			466	467	468	469	470	471	472	473	474	475	476	477	478				
			479	480	481	482	483	484	485	486	498	499	500	501	502				
			503	504	505	506	507	518	519	540	541	775	782	783	785				
			788	789	877	878	879	880	881	882	883	884	885	886	887				
			888	889	890	892	893	897	898	899	900	901	902	903	904				
			905	906	907	908	909	910	912	913	917	918	919	920	921				
			922	923	924	925	926	927	928	929	930	932	933	937	938				
			939	940	941	942	943	944	945	946	947	948	949	950	952				
			953	957	958	959	960	961	962	964	965	969	970	971	972				

973	974	976	977	981	982	983	984	985	986	988	989	993
994	995	996	997	998	999	1000	1044	1045	1046	1047	1048	1049
1050	1051	1052	1053	1054	1055	1056	1057	1058	1103	1104	1105	1106
1107	1108	1109	1261	1262	1274	1275	1276	1277	1282	1283	1284	1288
1289	1301	1302	1321	1322	1323	1325	1326	1327	1332	1333	1349	1350
1355	1356	1357	1358	1359	1360	1361	1362	1363	1364	1365	1366	1368
1369	1371	1372	1378	1379	1380	1381	1382	1383	1384	1385	1386	1387
1388	1389	1391	1392	1394	1395	1401	1402	1403	1404	1405	1406	1407
1408	1409	1410	1411	1412	1414	1415	1417	1418	1424	1425	1426	1427
1428	1429	1430	1431	1432	1433	1434	1435	1437	1438	1440	1441	1445
1446	1447	1448	1449	1450	1451	1452	1453	1454	1455	1456	1458	1459
1461	1462	1468	1469	1470	1471	1472	1473	1474	1475	1476	1477	1478
1479	1481	1482	1484	1485	1491	1492	1493	1494	1495	1496	1497	1498
1499	1500	1501	1502	1504	1505	1507	1508	1514	1515	1516	1517	1518
1519	1520	1521	1522	1523	1524	1525	1527	1528	1530	1531	1541	1542
1543	1544	1545	1546	1547	1548	1549	1550	1551	1552	1554	1555	1557
1558	1570	1571	1572	1573	1574	1575	1576	1577	1578	1579	1580	1581
1583	1584	1587	1588	1599	1600	1601	1603	1604	1609	1610	1611	1612
1613	1614	1615	1616	1617	1618	1619	1620	1622	1623	1625	1626	1632
1633	1634	1635	1636	1637	1638	1639	1640	1641	1642	1643	1645	1646
1649	1650	1656	1657	1658	1659	1660	1661	1662	1663	1664	1665	1666
1667	1669	1670	1672	1673	1679	1680	1681	1682	1683	1684	1685	1686
1687	1688	1689	1690	1692	1693	1695	1696	1702	1703	1704	1705	1706
1707	1708	1709	1710	1711	1712	1713	1715	1716	1717	1718	1724	1725
1726	1727	1728	1729	1730	1731	1732	1733	1734	1735	1737	1738	1739
1740	1746	1747	1748	1749	1750	1751	1752	1753	1754	1755	1756	1757
1759	1760	1762	1763	1769	1770	1771	1772	1773	1774	1775	1776	1777
1778	1779	1780	1782	1783	1785	1786	1796	1797	1798	1799	1800	1801
1802	1803	1804	1805	1806	1807	1809	1810	1812	1813	1825	1826	1827
1828	1829	1830	1831	1832	1833	1834	1835	1836	1838	1839	1841	1842
1860	1861	1862	1864	1865	1866	1867	1868	1869	1870	1871	1872	1874
1875	1876	1879	1880	1881	1884	1885	1886	1887	1888	1889	1890	1893
1894	1896	1897	1899	1900	1901	1904	1905	1906	1909	1910	1911	1912
1913	1914	1915	1918	1919	1925	1926	1927	1929	1930	1932	1933	1934
1937	1938	1939	1942	1943	1944	1945	1946	1947	1948	1951	1952	1954
1955	1957	1958	1959	1962	1963	1964	1967	1968	1969	1970	1971	1972
1973	1976	1977	1981	1982	1983	1985	1986	1988	1989	1990	1991	1992
1995	1996	1997	2000	2001	2002	2003	2004	2005	2006	2009	2010	2011
2012	2013	2020	2021	2027	2028	2029	2030	2031	2032	2033	2034	2035
2036	2037	2038	2039	2041	2042	2046	2047	2048	2049	2050	2051	2052
2053	2054	2055	2056	2057	2058	2060	2061	2065	2066	2067	2068	2069
2070	2071	2072	2073	2074	2075	2076	2077	2079	2080	2084	2085	2086
2087	2088	2089	2090	2091	2092	2093	2094	2095	2096	2098	2099	2103
2104	2105	2106	2107	2108	2109	2110	2111	2112	2113	2114	2115	2117
2118	2166	2167	2179	2180	2184	2185	2186	2187	2190	2191	2192	2197
2198	2202	2203	2207	2208	2212	2213	2214	2217	2218	2231	2232	2233
2234	2242	2243	2244	2245	2246	2247	2248	2249	2250	2251	2274	2275
2276	2277	2278	2279	2280	2286	2287	2288	2289	2290	2291	2292	2302
2303	2304	2305	2306	2307	2308	2313	2314	2315	2316	2317	2318	2320
2321	2322	2330	2331	2332	2333	2334	2335	2336	2337	2339	2340	2367
2368	2379	2380	2381	2382	2383	2384	2387	2388	2389	2390	2391	2392
2393	2399	2400	2401	2402	2403	2404	2405	2406	2407	2419	2420	2421
2422	2423	2424	2425	2426	2427	2428	2429	2430	2436	2437	2438	2439
2440	2441	2442	2444	2445	2448	2449	2453	2454	2455	2456	2457	2458
2459	2460	2461	2462	2463	2464	2470	2471	2472	2473	2474	2475	2476
2478	2479	2481	2482	2488	2489	2493	2494	2504	2505	2509	2510	2522

2523	2524	2525	2526	2527	2531	2532	2533	2534	2535	2536	2540	2541
2542	2543	2544	2545	2550	2551	2552	2553	2554	2555	2556	2557	2558
2559	2560	2561	2562	2563	2564	2565	2566	2569	2570	2571	2572	2573
2574	2575	2579	2580	2584	2585	2586	2592	2593	2594	2595	2596	2597
2598	2600	2601	2605	2606	2607	2608	2609	2610	2611	2613	2614	2618
2619	2620	2621	2622	2623	2624	2626	2627	2631	2632	2633	2634	2635
2636	2637	2639	2640	2644	2645	2646	2647	2648	2649	2652	2653	2761
2762	2789	2790	2793	2794	2795	2796	2797	2798	2799	2814	2815	2816
2817	2818	2819	2820	2824	2825	2826	2827	2828	2829	2830	2834	2835
2836	2837	2838	2839	2840	2845	2846	2847	2848	2849	2850	2851	2856
2857	2858	2859	2860	2861	2862	2867	2868	2869	2871	2872	2875	2876
2881	2882	2883	2884	2885	2886	2887	2892	2893	2894	2895	2896	2897
2898	2899	2900	2932	2933	2934	2935	2936	2937	2938	2939	2940	2941
2943	2944	2945	2946	2947	2948	2949	2951	2952	2953	2954	2955	2956
2957	2960	2961	2963	2964	2966	2967	2968	2969	2970	2971	2972	2973
2974	2979	2980	2981	2982	2983	2984	2985	2990	2991	2992	2993	2994
2995	2996	2997	2998	3030	3031	3032	3033	3034	3035	3036	3038	3039
3040	3041	3042	3043	3044	3046	3047	3048	3049	3050	3051	3052	3055
3056	3058	3059	3061	3062	3063	3064	3065	3066	3067	3068	3069	3075
3076	3077	3078	3079	3080	3081	3086	3087	3088	3089	3090	3091	3092
3093	3094	3122	3123	3124	3125	3126	3127	3128	3129	3130	3131	3133
3134	3135	3136	3137	3138	3139	3141	3142	3143	3144	3145	3146	3147
3150	3151	3153	3154	3156	3157	3158	3159	3160	3161	3162	3163	3164
3170	3171	3172	3173	3174	3175	3176	3181	3182	3183	3184	3185	3186
3187	3188	3189	3217	3218	3219	3220	3221	3222	3223	3224	3225	3226
3227	3228	3229	3230	3231	3232	3233	3234	3235	3236	3237	3238	3239
3240	3241	3242	3243	3244	3245	3246	3247	3248	3249	3251	3252	3254
3255	3256	3269	3270	3271	3272	3273	3274	3275	3279	3280	3284	3285
3286	3287	3288	3289	3292	3293	3294	3295	3296	3297	3298	3299	3301
3302	3306	3307	3308	3309	3310	3311	3315	3316	3385	3386	3412	3413
3416	3417	3418	3419	3420	3421	3422	3423	3424	3425	3426	3427	3428
3429	3430	3431	3432	3433	3434	3435	3436	3437	3438	3439	3440	3441
3442	3443	3445	3446	3467	3468	3481	3482	3484	3485	3486	3487	3488
3489	3490	3491	3492	3493	3494	3495	3496	3497	3498	3499	3500	3501
3502	3503	3504	3505	3506	3507	3508	3509	3510	3511	3512	3513	3514
3515	3516	3517	3518	3519	3520	3521	3522	3523	3524	3525	3526	3527
3528	3529	3530	3531	3533	3534	3568	3569	3573	3574	3575	3576	
SVCSUB= 177777	380#	1349	1371	1394	1417	1440	1461	1484	1507	1530	1557	1625
		1649	1672	1695	1717	1739	1762	1785	1812	1864	1896	1985
SVCTAG= 000000	2379	2448	2481	2493	2509	2789	2875	2963	3058	3153		
	380#	382#	530	531	579	580	891	892	911	912	931	951
	952	963	964	975	976	987	988	1058	1059	1260	1261	1288
	1300	1301	1331	1332	1367	1368	1390	1391	1413	1414	1436	1457
	1458	1480	1481	1503	1504	1526	1527	1553	1554	1582	1583	1587
	1621	1622	1644	1645	1668	1669	1691	1692	1714	1715	1736	1758
	1759	1781	1782	1808	1809	1837	1838	1840	1841	1892	1893	1918
	1950	1951	1975	1976	2008	2009	2019	2020	2040	2041	2059	2078
	2079	2097	2098	2116	2117	2165	2166	2216	2217	2338	2339	2367
	2393	2394	2407	2408	2443	2444	2477	2478	2487	2488	2503	2578
	2579	2599	2600	2612	2613	2625	2626	2638	2639	2651	2652	2761
	2820	2821	2830	2831	2840	2841	2851	2852	2862	2863	2870	2887
	2888	2900	2901	2959	2960	2974	2975	2985	2986	2998	2999	3055
	3069	3070	3081	3082	3094	3095	3149	3150	3164	3165	3176	3189
	3190	3250	3251	3278	3279	3300	3301	3314	3315	3384	3385	3469
SVCTST= 177777	3569	3570										
	380#	1347	1595	1854	2174	2229	2377	2787				

PARAMETER CODING
CVMBAD.P11

10-JUL-81

MACY11 30(1046)
17:08

20-JUL-81 11:44 PAGE 89
CROSS REFERENCE TABLE -- USER SYMBOLS

K 7

SEQ 0088

SWCHON	011304	2252*	2261*	2263*	2265*	2270*	2271*	2272*	2274	2281	2325#			
SWCNT	011300	2253*	2256*	2267*	2323#									
SWERR	011306 G	2245	2328#											
SWSET	002276	692#	1279*	2236*	2237	2331								
SYEE	016530	3174	3375#											
SYETST	015554	3198#	3215											
SYRTST	015212	3103#	3120											
SYSE	016416	2860	3222	3232	3242	3357#								
SYSEIN	014064	2853	2855#											
SYSET	015370	2865	3154#											
SYSR	016403	2849	3127	3135	3143	3354#								
SYSRIN	014036	2842	2844#	3027										
SYSROM	002370	707#	3072	3101	3167	3196								
SYSRT	015024	2854	3059#											
S\$LSYM=	010000	380#	531#	580#	892#	912#	932#	952#	964#	976#	988#	1051	1058	1059#
		1261#	1288#	1301#	1332#	1368#	1391#	1414#	1437#	1458#	1481#	1504#	1527#	1554#
		1583#	1587#	1622#	1645#	1669#	1692#	1715#	1737#	1759#	1782#	1809#	1838#	1841#
		1893#	1918#	1951#	1976#	2009#	2020#	2041#	2060#	2079#	2098#	2117#	2166#	2217#
		2339#	2367#	2388	2393	2394#	2400	2407	2408#	2444#	2478#	2488#	2504#	2579#
		2600#	2613#	2626#	2639#	2652#	2761#	2815	2820	2821#	2825	2830	2831#	2835
		2840	2841#	2846	2851	2852#	2857	2862	2863#	2871#	2882	2887	2888#	2893
		2900	2901#	2960#	2967	2974	2975#	2980	2985	2986#	2991	2998	2999#	3055#
		3062	3069	3070#	3076	3081	3082#	3087	3094	3095#	3150#	3157	3164	3165#
		3171	3176	3177#	3182	3189	3190#	3251#	3279#	3301#	3315#	3385#	3469#	3570#
TABLES	012672	2510	2656#											
TABL1	013134	2522	2692#											
TABL2	013233	2531	2704#											
TABL3	013332	2540	2716#											
TAG1	011136	2283#	2295											
TAG2	011166	2284	2292#											
TAG3	011212	2299#	2311											
TAG4	011242	2300	2308#											
TAG4A	011254	2297	2312#											
TEMP	011302	2251*	2257	2266*	2324#									
TSTCKW	005022	1144	1154	1158#										
T\$ARGC=	000001	402#	403#	404#	405#	406#	407#	877#	881	883#	889	897#	901	903#
		909	917#	921	923#	929	937#	941	943#	949	957#	961	969#	973
		981#	985	993#	999	1044#	1048	2027#	2031	2033#	2038	2046#	2050	2052#
		2057	2065#	2069	2071#	2076	2084#	2088	2090#	2095	2103#	2107	2109#	2114
		2274#	2279	2286#	2291	2302#	2307	2313#	2317	2330#	2336	2522#	2526	2531#
		2535	2540#	2544	2561#	2565	2569#	2574	2592#	2597	2605#	2610	2618#	2623
		2631#	2636	2644#	2648	3269#	3274	3284#	3288	3292#	3298	3306#	3310	
T\$CODE=	111004	1053#	2390#	2402#	2817#	2827#	2837#	2848#	2859#	2884#	2895#	2969#	2982#	2993#
		3064#	3078#	3089#	3159#	3173#	3184#	3416#	3422#	3428#	3434#	3440#	3445#	3484#
		3490#	3496#	3502#	3508#	3514#	3520#	3526#	3533#					
T\$ERRN=	000062	380#	1104#	1356#	1379#	1402#	1425#	1446#	1469#	1492#	1515#	1542#	1571#	1610#
		1633#	1657#	1680#	1703#	1725#	1747#	1770#	1797#	1826#	1885#	1910#	1943#	1968#
		2001#	2243#	2420#	2437#	2454#	2471#	2555#	2936#	2944#	2952#	3031#	3039#	3047#
		3126#	3134#	3142#	3221#	3231#	3241#							
T\$EXCP=	000000	1053#	1058	2402#	2407	2895#	2900	2969#	2974	2993#	2998	3064#	3069	3089#
		3094	3159#	3164	3184#	3189	3416#	3421	3422#	3427	3428#	3433	3434#	3439
		3484#	3489	3490#	3495	3496#	3501	3502#	3507	3508#	3513	3514#	3519	3520#
		3525	3526#	3531										
T\$FLAG=	000041	1325#	1362#	1385#	1408#	1431#	1452#	1475#	1498#	1521#	1548#	1577#	1599#	1616#
		1639#	1663#	1686#	1709#	1731#	1753#	1776#	1803#	1832#	1860#	1925#	1981#	2011#
		2190#	2212#	2249#	2320#	2426#	2460#	2584#	2793#	2867#	2932#	3122#	3217#	3227#

PARAMETER CODING
CVM8AD.P11 10-JUL-81

MACY11 30(1046)

20-JUL-81 11:44 PAGE 90
CROSS REFERENCE TABLE -- USER SYMBOLS

L 7

SEQ 0089

T\$GMAN= 000000	3237# 380#	3247# 1050#	3254# 1059#	3445# 2399#	3533# 2408#	2892# 2901#	2966# 2975#	2990# 2999#	3061# 3070#				
T\$HILI= 177777	3086# 1053# 3093# 3484# 3524# 380#	3095# 1057# 3159# 3488# 3526# 3574#	3156# 2402# 3163# 3490# 3530#	3165# 2406# 3184# 3494#	3181# 2895# 3188# 3496#	3190# 2899# 3416# 3500#	2969# 2973# 3420# 3502#	2993# 2997# 3422# 3506# 3508# 3512#	3064# 3068# 3432# 3514#	3068# 3434# 3432# 3518#	3089# 3438# 3438# 3520#		
T\$LAST= 000001	3092# 3484# 3523# 380#	3159# 3487# 3526# 531#	3162# 3490# 3529# 580#	3184# 3493# 892# 912#	3187# 3496# 912# 932#	3416# 3499# 932# 952#	3419# 3502# 952# 964#	3425# 3508# 976# 988#	3428# 3511# 988# 1261#	3431# 3514# 1261# 1288#	3434# 3517# 1288# 1301#	3089# 3437# 3520# 1301#	
T\$LOLI= 000000	3092# 3484# 3523# 380#	3159# 3487# 3526# 531#	3162# 3490# 3529# 580#	3184# 3493# 892# 912#	3187# 3496# 912# 932#	3416# 3499# 932# 952#	3419# 3502# 952# 964#	3425# 3508# 976# 988#	3428# 3511# 988# 1261#	3431# 3514# 1261# 1288#	3434# 3517# 1288# 1301#	3089# 3437# 3520# 1301#	
T\$LSYM= 010000	1332# 1645# 1976# 2478# 3150# 3577# 380#	1368# 1669# 2009# 2488# 3251# 391#	1391# 1692# 2020# 2504# 3279# 487#	1414# 1715# 2041# 2579# 3301# 497#	1437# 1737# 2060# 2600# 3315# 508#	1458# 1759# 2079# 2613# 3385# 518#	1481# 1782# 2098# 2626# 3469# 530#	1504# 1809# 2117# 2639# 3570# 540#	1527# 1838# 2166# 2652# 579#	1554# 1841# 2217# 2761# 595#	1583# 1893# 2339# 2871# 668#	1587# 1918# 2367# 2960# 677#	1622# 1951# 2444# 3055# 709#
T\$LTNO= 000007	380#	391#	487#	497#	508#	518#	530#	540#	579#	595#	668#	677#	709#
T\$NEST= 177777	875# 987# 1367# 1507# 1668# 1812# 2008# 2165# 2487# 2642# 3250# 3568#	891# 1258# 1371# 1526# 1672# 1837# 2016# 2174# 2493# 2651# 3267#	895# 1260# 1390# 1530# 1691# 1840# 2019# 2216# 2503# 2760# 3278#	911# 1272# 1394# 1553# 1695# 1854# 2025# 2229# 2509# 2787# 3282#	915# 1287# 1413# 1557# 1714# 1864# 2040# 2328# 2578# 2789# 3300#	931# 1291# 1417# 1582# 1717# 1892# 2044# 2338# 2590# 2870# 3304#	935# 1296# 1436# 1586# 1736# 1896# 2059# 2366# 2599# 2875# 3314#	951# 1298# 1440# 1595# 1739# 1917# 2063# 2377# 2603# 2959# 3384#	955# 1300# 1457# 1603# 1758# 1929# 2078# 2379# 2612# 2963# 3412#	963# 1310# 1461# 1621# 1762# 1950# 2082# 2443# 2616# 3054# 3445#	967# 1331# 1480# 1625# 1781# 1954# 2097# 2448# 2625# 3058# 3467#	975# 1347# 1484# 1644# 1785# 1975# 2101# 2477# 2629# 3149# 3481#	979# 1349# 1503# 1649# 1808# 1985# 2116# 2481# 2638# 3153# 3533#
T\$NSO = 000005	391# 891# 1258# 1840# 3467#	487# 895# 1260# 1854# 3481#	497# 911# 1272# 2165# 3533#	508# 915# 1287# 2174# 3568#	518# 931# 1291# 2216#	530# 935# 1296# 2229#	540# 951# 1298# 2366#	579# 955# 1300# 2377#	595# 963# 1310# 2760#	668# 967# 1331# 2787#	677# 975# 1347# 3384#	709# 979# 1586# 3412#	875# 987# 1595# 3445#
T\$NS1 = 000011	1349# 1503# 1672# 1837# 2025# 2443# 2616# 3149#	1367# 1507# 1691# 1864# 2040# 2448# 2625# 3153#	1371# 1526# 1695# 1892# 2044# 2477# 2629# 3250#	1390# 1530# 1714# 1896# 2059# 2481# 2638# 3267#	1394# 1553# 1717# 1917# 2063# 2487# 2642# 3278#	1413# 1557# 1736# 1929# 2078# 2493# 2651# 3282#	1417# 1582# 1739# 1950# 2082# 2503# 2789# 3300#	1436# 1603# 1758# 1954# 2097# 2509# 2870# 3304#	1440# 1621# 1762# 1975# 2101# 2578# 2875# 3314#	1457# 1625# 1781# 1985# 2116# 2590# 2959#	1461# 1644# 1785# 2008# 2116# 2599# 2963#	1480# 1649# 1808# 2016# 2328# 2603# 3054#	1484# 1668# 1812# 2019# 2379# 2612# 3058#
T\$PTNU= 000000	380#												
T\$SAVL= 177777	380#												
T\$SEGL= 177777	380#												
T\$SUBN= 000005	380#	1347#	1349#	1371#	1394#	1417#	1440#	1461#	1484#	1507#	1530#	1557#	1595#
	1603#	1625#	1649#	1672#	1695#	1717#	1739#	1762#	1785#	1812#	1854#	1864#	1896#
	1929#	1954#	1985#	2174#	2229#	2377#	2379#	2448#	2481#	2493#	2509#	2787#	2789#
	2875#	2963#	3058#	3153#									
T\$TAGL= 177777	380#												
T\$TAGN= 010111	380#	518#	540#	875#	895#	915#	935#	955#	967#	979#	1258#	1272#	1291#
	1298#	1310#	1347#	1349#	1371#	1394#	1417#	1440#	1461#	1434#	1507#	1530#	1557#
	1595#	1603#	1625#	1649#	1672#	1695#	1717#	1739#	1762#	1785#	1812#	1854#	1864#

PARAMETER CODING
CVMBAD.P11

10-JUL-81 17:08

MACY11 30(1046)

20-JUL-81 11:44 PAGE 93
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0092

T7	013646 G	506	2787#																
T7.1	013646	2789#																	
T7.2	014120	2875#																	
T7.3	014454	2963#																	
T7.4	015022	3058#																	
T7.5	015366	3153#																	
UAM =	000200 G	655#																	
ULIMIT	002270	689#	1090*	1092	1094	1096*	1099*	1100											
UNIT	016646	3417	3447#																
UNKNWN	013431	2561	2728#																
VECT	002274	691#	1277*	1321	1868														
VIPRI	004230	992#	3276	3290	3312														
VIRMSG	003605	849#	995																
VIRTAD	004504	1089#	2649	2915	3013	3109	3204	3275	3289	3311									
VRTPCR	002242	678#	2410*	2411	2482*	2483	2495*	2497	2499*										
WDELAY	005212	1222#	1876	1901	1934	1959	1992	2176	2181	2194	2199	2204	2209						
WORD	015756	2968	2975	3004	3263#														
WORDCT	002302	694#	1060*	2877*	2975*	3071*	3166*												
X\$ALWA=	000000	380#	3445	3533															
X\$FALS=	000040	380#																	
X\$OFFS=	000400	380#	3445	3533															
X\$TRUE=	000020	380#																	
ZERR	003174	796#	877																
.	= 017424	2#	705#	706#	707#	1326	1363	1386	1409	1432	1453	1476	1499	1522					
		1549	1578	1600	1617	1640	1664	1687	1710	1732	1754	1777	1804	1833					
		1861	1926	1982	2012	2191	2213	2250	2321	2364#	2427	2461	2585	2654#					
		2758#	2794	2868	2933	3123	3218	3228	3238	3248	3255	3445	3533	3564#					
		3571#																	

. ABS. 017424 000

ERRORS DETECTED: 0

CVMBAD, CVMBAD/SOL/CRF:SYM/NL:TOC=SVC/ML, CVMBAD.P11

RUN-TIME: 31 30 2 SECONDS

RUN-TIME RATIO: 205/64=3.1

CORE USED: 18K (36 PAGES)