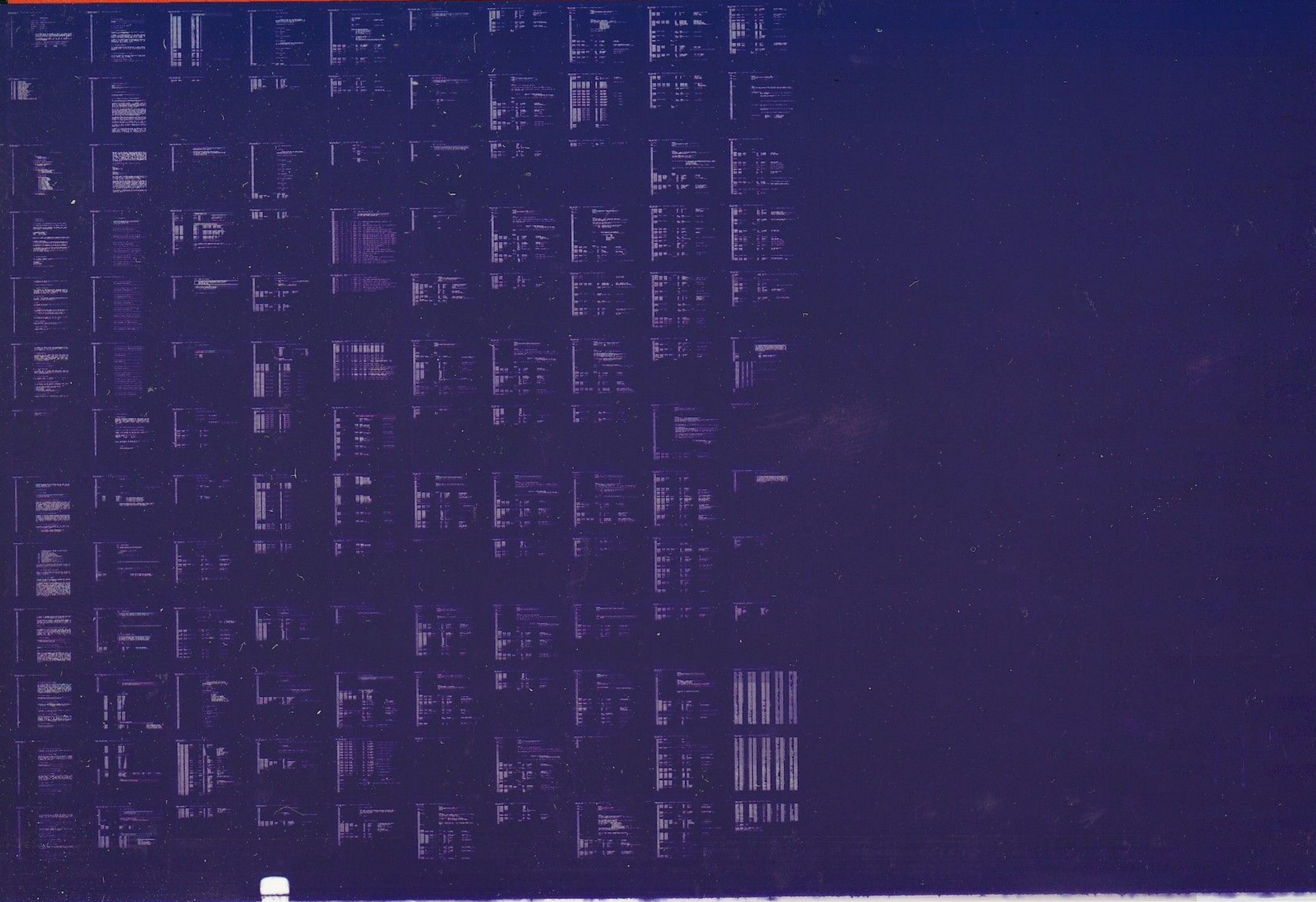KMV11-A,
KMV11-B

LOGIC DIAG
CVKMAAO

AH-T370A-MC

FICHE 1 OF 1

MAY 1983
COPYRIGHT© 82-83
MADE IN USA

digital

```
   5                          .REM @
   6
   7
   8                                    IDENTIFICATION
   9                                    ---------------
  10
  11                          PRODUCT CODE:   AC-T369A-MC
  12
  13                          PRODUCT NAME:   CVKMAA0 KMV11A/B LOGIC DIAG
  14
  15                          PRODUCT DATE:   JAN 1983
  16
  17                          MAINTAINER:     CSS ANNECY
  18
  19                          AUTHOR:         MICHELET, GUY
  20
  21
  22
  23
  24
  25                          THE INFORMATION IN THIS DOCUMENT IS  SUBJECT   TO  CHANGE WITHOUT
  26                          NOTICE  AND  SHOULD  NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL
  27                          EQUIPMENT CORPORATION.  DIGITAL EQUIPMENT CORPORATION ASSUMES NO
  28                          RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS DOCUMENT.
  29
  30                          NO RESPONSIBILITY IS ASSUMED  FOR  THE  USE  OR  RELIABILITY  OF
  31                          SOFTWARE ON EQUIPMENT THAT IS NOT  SUPPLIED  BY  DIGITAL  OR ITS
  32                          AFFILIATED COMPANIES.
  33
  34                          COPYRIGHT (C) 1982,1983 BY DIGITAL EQUIPMENT CORPORATION
  35
  36                          THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:
  37
  38                              DIGITAL        PDP           UNIBUS          MASSBUS
  39                              DEC            DECUS         DECTAPE
```

KMV11 A/B LOGIC DIAG   MACRO M1200 06-JAN-83 09:39
TABLE OF CONTENTS

```
41
42
43
44
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
```

TABLE OF CONTENTS

```
 94
 95
 96
 97
 98
 99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
```

1.0 INTRODUCTION

1.1 PROGRAM ABSTRACT

THIS DIAGNOSTIC WAS DESIGNED TO TEST OUT THE KMV11 MODULE

THE PROGRAM WAS IMPLEMENTED USING THE DIAGNOSTIC SUPERVISOR.

THROUGH DIALOGUE WITH THE OPERATOR, THE PROGRAM WILL ALLOW
MODIFICATION OF DEVICE PARAMETERS, SUCH AS UNIBUS ADDRESS,
VECTOR ADDRESS, AND PRIORITY LEVEL.

1.2 HARDWARE INTRODUCTION
HARDWARE DESCRIPTION:
M7500   = KMV11-A MODULE
M7501   = KMV11-B MODULE

KMV11-A IS A SINGLE LINE COMMUNICATION CONTROLER FOR QBUS SYSTEMS
KMV11-B IS A DUAL LINE COMMUNICATION CONTROLER FOR QBUS SYSTEMS

DIAGNOSTIC DECRIPTION:
THE KMV11 STATIC DIAGNOSTIC IS COMPATIBLE WITH BOTH KMV11 A/B
IT WILL RUN IN STAND ALONE WITHOUT ANY OPERATOR INTERVENTIONS

THE PURPOSE OF THIS DIAGNOSTIC IS TO TEST ALL THE HARDWARE OF
THE QBUS PART OF THE INTERFACE AND THE RAM PART OF THE KMV11.

THIS DIAGNOSTIC WILL FIRST TEST QBUS ACCESS ON KMV11A(M7500) AND
KMV11B(M7501) CSR'S REGISTERS, THEN DATA TRANSFER FROM QBUS
TO DCT11 MICROPROCESSOR.
AFTER THAT IT WILL TEST KMV11 RAM MEMORY, DMA TRANSFERS IN/OUT
KMV11 AND INTERUPT CAPABILITY.

2.0 HARDWARE REQUIREMENTS

THE FOLLOWING HARDWARE IS REQUIRED TO RUN THE STATIC LOGIC
TESTS ON MODULES M7500 OR M7501:

PDP-11/03,23,23 PLUS
16K MEMORY
CONSOLE TERMINAL
REAL TIME CLOCK

3.0 PRELIMINARY PROGRAM REQUIREMENTS

```
151
152                        THE PROCESSOR AND MEMORY SHOULD BE THOROUGHLY   TESTED   PRIOR
153                        TO RUNNING THIS DIAGNOSTIC.
154
155
156                        4.0 GENERAL PROGRAM CONSIDERATIONS
157
158
159                        4.1 DIAGNOSTIC SUPERVISOR
160
161                        THIS  PROGRAM  IS  COMPATIBLE WITH THE STANDALONE DIAGNOSTIC
162                        SUPERVISOR, AND MUST BE LOADED TO BE  CO-RESIDENT  WITH  THE
163                        SUPERVISOR,  OR  BE  PREVIOUSLY COMBINED WITH THE SUPERVISOR
164                        AND LOADED AS A SINGLE FILE.  IN EITHER CASE,  THE  COMBINED
165                        PROGRAM WILL NOT EXCEED 16K OF MEMORY.
166
167
168                        4.2 EXECUTION TIME
169
170                        THE  TOTAL  TIME  REQUIRED  TO RUN THE M7500 OR M7501 STATIC
171                        DIAGNOSTIC  IS ABOUT 120 SECONDS PER PASS FOR EACH UNIT (ON
172                        PDP11/23A WITH SUPERVISOR VERSION C4).
173
174
175                        4.3 XXDP+
176
177                        THIS PROGRAM MAY BE LOADED UNDER XXDP+, AND MAY  BE  RUN  IN
178                        DUMP MODE OR CHAIN MODE.
179
180
181                        4.4 ACT/SLIDE
182
183                        THIS PROGRAM MAY BE LOADED UNDER ACT OR SLIDE AND MAY BE RUN
184                        IN DUMP MODE (FOR THAT DIAGNOSTIC MUST BE SETUP FIRST).
185
186                        CAUTION:UNDER SLIDE THE OPERATOR MUST ALWAYS ANSWER ''YES''
187                                (THE FIRST TIME)FOR HARDWARE PRAMETERS CHANGE.
188
189                        4.5 APT
190
191                        THIS PROGRAM MAY BE LOADED  BY  THE  APT  SYSTEM  (INCLUDING
192                        APT-RD) AND RUN IN PROGRAM MODE OR SCRIPT MODE.
193
194
195                        4.6 MEMORY MANAGEMENT
196
197                        MEMORY MANAGEMENT IS NOT UTILIZED IN THIS PROGRAM.  IF IT IS
198                        INSTALLED, IT IS DISABLED BY THE PROGRAM.
199
200
201                        4.7 MEMORY PARITY OPTION
202
203                        IF PARITY MEMORY  IS  INSTALLED,  MEMORY  PARITY  TRAPS  ARE
204                        DISABLED BY THE PROGRAM.
205
206
207                        4.8 ERROR LOGGING
```

208
209   THE NUMBER OF ERRORS WHICH HAVE OCCURRED ON EACH DEVICE
210   UNDER TEST SINCE THE LAST START OR RESTART COMMAND IS KEPT
211   IN AN ERROR LOG.   THIS LOG MAY BE PRINTED BY USING THE
212   ''PRINT'' COMMAND (SEE SECTION 6.3.8).
213
214
215   5.0 PROGRAM LOAD MEDIA
216
217   THIS PROGRAM CAN BE LOADED FROM PAPER TAPE USING THE
218   ABSOLUTE LOADER OR FROM ACT, SLIDE, OR APT SYSTEMS, OR FROM
219   ANY MEDIA SUPPORTED BY XXDP+. WHEN USING THE PAPER TAPE
220   ABSOLUTE   LOADER,   THE PROGRAM SHOULD BE LOADED FIRST,
221   FOLLOWED BY THE DIAGNOSTIC SUPERVISOR.   WHEN USING XXDP+,
222   THE DIAGNOSTIC SUPERVISOR SHOULD BE LOADED FIRST, FOLLOWED
223   BY THE DIAGNOSTIC PROGRAM.
224
225
226   6.0 OPERATING INSTRUCTIONS
227
228
229   6.1 LOADING AND STARTING PROCEDURES
230
231
232   6.1.1 LOADING PROCEDURES
233
234   THIS PROGRAM MAY BE LOADED FROM PAPER TAPE USING THE
235   ABSOLUTE LOADER. IT MAY ALSO BE LOADED FROM ANY XXDP+ LOAD
236   MEDIA. WHEN LOADED UNDER XXDP+, THE DIAGNOSTIC SUPERVISOR
237   WILL BE LOADED AUTOMATICALLY.
238
239
240   6.1.2 STARTING PROCEDURES
241
242   THE PROGRAM STARTS AT LOCATION 200. USE STANDARD DEC
243   PROCEDURES TO START THE PROGRAM.
244
245
246   6.1.3 STEPS FOR QUICK AND SIMPLE EXECUTION
247
248   THE DIAGNOSTIC CAN BE EXECUTED STANDALONE UNDER XXDP+
249   WITHOUT READING THE REMAINDER OF THIS DOCUMENT, AS FOLLOWS:
250
251   A) LOAD AND START DIAGNOSTIC USING RUN COMMAND
252   B) RECEIVE DIAGNOSTIC SUPERVISOR PROMPT   (DR>)
253   C) ENTER STA<CR>
254   D) ANSWER HARDWARE QUESTIONS
255   E) GET END OF PASS MESSAGES OR ERROR MESSAGES
256   F) TO END EXECUTION, ENTER CONTROL/C
257
258
259   6.2 INITIAL DIALOGUE
260
261   AFTER THE PROGRAM AND THE SUPERVISOR ARE LOADED AND THE
262   PROGRAM IS STARTED , THE FOLLOWING IDENTIFICATION IS TYPED:
263
264   DRS LOADED

KMV11A/B LOGIC DIAG   MACRO M1200 06-JAN-83 09:39 PAGE 4-3
PROGRAM DOCUMENT

```
265                          DIAG.  RUN-TIME SERVICES
266                          VKMAA0-A-0
267                          KMV11A/B LOGIC DIAGNOSTIC
268                          DR>
```

```
270
271
272
273
274
275         THE OPERATOR THEN PROCEEDS BY TYPING  ONE  OR  MORE  OF  THE
276         COMMANDS  DESCRIBED  IN  THE FOLLOWING SECTION 6.3.(FOR MORE
277         DETAILED INFORMATION, REFER  TO  THE  DIAGNOSTIC  SUPERVISOR
278         FUNCTIONAL SPECIFICATION).
279
280
281         6.3 PROGRAM OPTIONS
282
283
284         6.3.1 START COMMAND
285
286         *****************************************************************
287         STA(RT)/TESTS:<TEST-LIST>/PASS:<PASS-CNT>/FLAGS:
288            <FLAG-LIST>/EOP:<INCR>
289         *****************************************************************
290
291
292         6.3.1.1 TESTS SWITCH (/TESTS:<TEST-LIST>)
293
294         <TEST-LIST> IS A SEQUENCE OF DECIMAL NUMBERS (1:2  ETC.)  OR
295         RANGES  OF  DECIMAL NUMBERS (1-5:8-10 ETC.) THAT SPECIFY THE
296         TESTS TO BE EXECUTED.  THE NUMBERS ARE SEPARATED BY  COLONS.
297         THE  NUMBERS  RANGE FROM 1 TO THE LARGEST TEST NUMBER IN THE
298         DIAGNOSTIC.  THEY MAY BE SPECIFIED IN ANY ORDER.  TESTS WILL
299         BE  EXECUTED  IN  NUMERICAL ORDER REGARDLESS OF THE ORDER OF
300         SPECIFICATION.  THE DEFAULT IS TO  EXECUTE  ALL  TESTS.   ON
301         THIS AND ALL SWITCHES, THE ANGLE BRACKETS <> ARE PUNCTUATION
302         USED IN THE DEFINITION ONLY, AND ARE NOT TO BE TYPED BY  THE
303         OPERATOR.  SEE EXAMPLE AT END OF 6.3.1.5.
304
305
306         6.3.1.2 PASS SWITCH (/PASS:<PASS-CNT>)
307
308         <PASS-CNT> IS A DECIMAL NUMBER INDICATING THE DESIRED NUMBER
309         OF PASSES.  A PASS IS DEFINED AS THE EXECUTION OF  THE  FULL
310         DIAGNOSTIC (ALL SELECTED TESTS) AGAINST ALL UNITS SUBMITTED.
311         THE DEFAULT IS NON-ENDING EXECUTION.  IN THIS CASE EXIT FROM
312         THE  PROGRAM IS ACCOMPLISHED EITHER BY TYPING A CONTROL/C OR
313         BY OCCURANCE OF AN ERROR WITH THE HALT ON ERROR  FLAG  BEING
314         SET.   THE EXIT IS A RETURN TO COMMAND MODE.  SEE EXAMPLE AT
315         END OF 6.3.1.5.
316
317
318         6.3.1.3 FLAGS SWITCH (/FLAGS:<FLAG-LIST>)
319
320         <FLAG-LIST> IS A SEQUENCE OF ELEMENTS OF  THE  FORM  <FLAG>,
321         <FLAG=1>, OR <FLAG=0>, SEPARATED BY COLONS, WHERE <FLAG> HAS
322         ONE OF THE FOLLOWING VALUES:
323
324            HOE   HALT ON ERROR, CAUSING COMMAND MODE TO BE
325                  ENTERED WHEN AN ERROR IS ENCOUNTERED
326            LOE   LOOP ON ERROR, CAUSING THE DIAGNOSTIC TO LOOP
```

328
329
330
331
332
333
334   CONTINUOUSLY WITHIN THE SMALLEST DEFINED BLOCK
335   OF CODING (SEGMENT, SUBTEST, OR TEST) CONTAIN-
                                       ING THE ERROR
336   IER   INHIBIT ERROR REPORTING
337   IBE   INHIBIT BASIC ERROR REPORTS
338   IXE   INHIBIT EXTENDED ERROR REPORTS
339   PRI   DIRECT ALL MESSAGES TO A LINE PRINTER
340   PNT   PRINT NUMBER OF TEST BEING EXECUTED
341   BOE   BELL ON ERROR
342   UAM   RUN IN UNATTENDED MODE, BYPASSING MANUAL
343   INTERVENTION TESTS
344   ISR   INHIBIT STATISTICAL REPORTS
345   IDU   INHIBIT DROPPING OF UNITS BY DIAGNOSTIC
346   LOT   LOOP ON TEST
347
348   THE FLAGS NAMED OR EQUATED TO 1 ARE SET, THOSE EQUATED TO 0
349   ARE CLEARED. A FLAG NOT SPECIFIED IS CLEARED. IF THE FLAGS
350   SWITCH IS NOT GIVEN ALL FLAGS ARE CLEARED. SEE EXAMPLE AT
351   END OF 6.3.1.5.
352
353
354   6.3.1.4 END OF PASS SWITCH (/EOP:<INCR>)
355
356   <INCR> IS A DECIMAL NUMBER INDICATING HOW OFTEN (IN TERMS OF
357   PASSES) IT IS DESIRED THAT THE END OF PASS MESSAGE BE
358   PRINTED.   THE DEFAULT IS AT THE END OF EVERY PASS. SEE
359   EXAMPLE AT END OF 6.3.1.5.
360
361
362   6.3.1.5 EFFECT OF START COMMAND
363
364   THE EFFECT OF THE START COMMAND IS TO INITIATE THE HARDWARE
365   PARAMETER DIALOGUE, THE SOFTWARE PARAMETER DIALOGUE, AND
366   THEN THE DIAGNOSTIC TESTS THEMSELVES.
367
368   THE HARDWARE PARAMETER DIALOGUE COMMENCES WITH THE QUESTION
369   "# UNITS?" TO WHICH THE OPERATOR REPLIES WITH A DECIMAL
370   NUMBER N FROM 1 TO 16. THE TERM "UNIT" REFERS TO THE DEVICE
371   TO WHICH THIS SERIES OF DIAGNOSTICS IS DEDICATED. FOLLOWING
372   THIS ARE THE QUESTIONS WHEREBY THE P-TABLES THEMSELVES WILL
373   BE BUILT.   EACH P-TABLE IS A CORE-RESIDENT TABLE CONTAINING
374   ALL THE HARDWARE INFORMATION FOR ONE UNIT.   THE OPERATOR
375   MUST SUPPLY N (NUMBER OF UNITS) VALUES FOR EACH QUESTION.
376   HE MAY DO THIS BY GIVING ONE ANSWER TO EACH QUESTION (IN
377   WHICH CASE THE SERIES OF QUESTIONS WILL BE POSED N TIMES) OR
378   BY GIVING N VALUES, SEPARATED BY COMMAS, TO EACH QUESTION
379   (SERIES WILL BE POSED ONCE). EACH QUESTION IS FOLLOWED BY
380   THE RESPONSE RADIX (D FOR DECIMAL, B FOR BINARY, O FOR
381   OCTAL, L FOR YES/NO) IN PARENTHESES AND THE DEFAULT VALUE
382   AFTER THE PARENTHESES.

```
384
385
386
387
388
389            FOLLOWING THE HARDWARE QUESTIONS ARE THE SOFTWARE  QUESTIONS
390            TO  BUILD  THE SOFTWARE TABLES, WHICH DEFINE THE MODE (QUICK
391            VERIFY ETC.) THAT THE DIAGNOSTIC WILL EXECUTE IN.
392
393            WHEN THE QUESTION "# UNITS?" IS ANSWERED, MEMORY STORAGE  IS
394            ALLOCATED  FOR  THE  P-TABLES, AND IF THERE IS NOT ENOUGH TO
395            ACCOMMODATE THEM THE MESSAGE "TOO MANY UNITS" IS ISSUED.  IN
396            THIS  CASE THE DIAGNOSTIC MUST BE EXECUTED MORE THAN ONCE TO
397            TEST ALL UNITS.
398
399            EXAMPLE:
400
401            STA/TESTS:1:2-4:6:8-10/PASS:3/FLAGS:IER:HOE=1:UAM:LOE
402
403            THIS COMMAND WILL CAUSE THREE PASSES TO BE MADE,  EACH  PASS
404            CONSISTING  OF  TESTS 1,2,3,4,6,8,9, AND 10 EXECUTED AGAINST
405            ALL UNITS.  THERE IS NO DIFFERENCE BETWEEN SAYING <FLAG> AND
406            SAYING <FLAG=1>. THE NOTATION <FLAG=0> IS MEANINGFUL ONLY ON
407            A COMMAND  OTHER  THAN  START  TO  CLEAR  A FLAG THAT WAS
408            PREVIOUSLY  SET.   NOTE  THAT ON ALL COMMANDS ONLY THE FIRST
409            THREE LETTERS ARE SCANNED.
410
411
412            6.3.2 RESTART COMMAND
413
414            ******************************************************************
415            RES(TART)/TESTS:<TEST-LIST>/PASS:<PASS-CNT>/FLAGS:
416               <FLAG-LIST>/UNITS:<UNIT-LIST>
417            ******************************************************************
418
419
420            6.3.2.1 TESTS, PASS, AND FLAGS SWITCHES
421
422            <TEST-LIST>, <PASS-CNT>, AND <FLAG-LIST> ARE AS IN THE START
423            COMMAND.
424
425
426            6.3.2.2 UNITS SWITCH (/UNITS:<UNIT-LIST>)
427
428            <UNIT-LIST> IS A SEQUENCE OF DECIMAL NUMBERS (0,1  ETC.)  OR
429            RANGES  OF DECIMAL NUMBERS (0-5, 8-10 ETC.) THAT SPECIFY THE
430            UNITS TO BE TESTED.  THE NUMBERS ARE  SEPARATED  BY  COLONS.
431            THE  NUMBERS  MAY  RANGE FROM 0 THRU N-1 (N IS THE NUMBER OF
432            UNITS SPECIFIED IN THE PREVIOUS START COMMAND).  THE  NUMBER
433            INDICATES  THE  POSITION  OF  THE  P-TABLE  AS  THE DATA WAS
434            ENTERED DURING THE HARDWARE DIAGLOGUE.  THE UNITS WHICH  ARE
435            SELECTED  MUST  NOT  HAVE  BEEN DROPPED BY THE DROP COMMAND.
436            SEE THE DISCUSSION OF ADD AND DROP COMMANDS BELOW.   DEFAULT
437            IS  TO  TEST ALL UNITS WHICH HAVE NOT BEEN DROPPED BY A DROP
438            COMMAND.
```

```
440
441
442
443
444
445                    6.3.2.3 EFFECT OF RESTART COMMAND
446
447                    THE RESTART COMMAND DIFFERS FROM THE START COMMAND  IN  THAT
448                    THE  P-TABLES  FROM  THE  PREVIOUS START COMMAND (THERE MUST
449                    HAVE BEEN ONE) ARE USED, INSTEAD OF NEW  ONES  BEING  BUILT.
450                    THE  UNITS  SWITCH  GIVES  THE ABILITY TO SELECT A SUBSET OF
451                    THESE.  THE SOFTWARE DIALOGUE MAY OPTIONALLY  BE  REEXECUTED
452                    (OPERATOR  WILL  BE  ASKED).  THE  COMMAND CAN BE USED AFTER
453                    COMMAND MODE HAS BEEN REENTERED IN ANY OF THE  THREE  NORMAL
454                    WAYS:   A)  THE REQUESTED NUMBER OF PASSES HAVE BEEN MADE B)
455                    AN ERROR WAS ENCOUNTERED WITH THE HALT ON ERROR FLAG SET  C)
456                    A CONTROL/C WAS ENTERED BY THE OPERATOR.
457
458
459                    6.3.3 CONTINUE COMMAND
460
461                    *************************************************************
462                    CON(TINUE)/PASS:<PASS-CNT/FLAGS:<FLAG-LIST>
463                    *************************************************************
464
465
466                    6.3.3.1 PASS SWITC:: (/PASS:<PASS-CNT>)
467
468                    <PASS-CNT>  IS  SAME AS IN START COMMAND, BUT THE DEFAULT IS
469                    THE UNSATISFIED PASS-CNT FROM THE PREVIOUS START OR RESTART.
470                    IF NONE REMAINS, THE DEFAULT IS NON-ENDING EXECUTION.
471
472
473                    6.3.3.2 FLAG SWITCH (/FLAGS:<FLAG-LIST>)
474
475                    <FLAG-LIST>  IS  SAME  AS  IN START COMMAND, BUT UNSPECIFIED
476                    FLAGS RETAIN THEIR CURRENT VALUE.
477
478
479                    6.3.3.3 EFFECT OF CONTINUE COMMAND
480
481                    CONTINUE MUST FOLLOW A START OR RESTART,  AND  COMMAND  MODE
482                    MUST  HAVE  BEEN  ENTERED  DUE  TO  A  HALT  ON  ERROR  OR A
483                    CONTROL/C. THE EFFECT  OF  THE  COMMAND  IS  TO  GO  TO  THE
484                    BEGINNING  OF THE TEST THAT WAS BEING EXECUTED WHEN THE HALT
485                    OR CONTROL/C TOOK PLACE.  SOFTWARE DIALOGUE  MAY  OPTIONALLY
486                    BE REEXECUTED.  HARDWARE PARAMETERS MAY NOT BE CHANGED.
487
488
489                    6.3.4 PROCEED COMMAND
490
491                    *************************************************************
492                    PRO(CEED)/FLAGS:<FLAG-LIST>
493                    *************************************************************
```

```
495
496
497
498
499
500              6.3.4.1 FLAGS SWITCH (/FLAGS:<FLAG-LIST>)
501
502              <FLAG-LIST> IS AS IN THE  START COMMAND, BUT UNSPECIFIED
503              FLAGS RETAIN THEIR CURRENT VALUE.
504
505
506              6.3.4.2 EFFECT OF PROCEED COMMAND
507
508              PROCEED MUST FOLLOW A START, RESTART, OR CONTINUE.   COMMAND
509              MODE MUST HAVE BEEN ENTERED VIA A HALT ON ERROR. THE EFFECT
510              OF THE  COMMAND  IS  TO BEGIN  EXECUTION  AT  THE  LOCATION
511              FOLLOWING  THE  ERROR  CALL.   NEITHER HARDWARE NOR SOFTWARE
512              PARAMETERS MAY BE ALTERED.
513
514
515              6.3.5 ADD COMMAND
516
517              **************************************************************
518              ADD/UNITS:<UNIT-LIST>
519              **************************************************************
520
521
522              6.3.5.1 UNITS SWITCH (/UNITS:<UNIT-LIST>
523
524              <UNIT-LIST> IS AS IN THE RESTART COMMAND.
525
526
527              6.3.5.2 EFFECT OF ADD COMMAND
528
529              THE UNITS SPECIFIED ARE ADDED TO THE  TEST  SEQUENCE.  EACH
530              UNIT  MUST  HAVE  A  P-TABLE  IN  MEMORY  DUE  TO AN EARLIER
531              HARDWARE DIALOGUE. THIS  COMMAND  MUST  BE  FOLLOWED  BY  A
532              RESTART  OR  CONTINUE.   THE  UNITS SWITCH MUST BE SPECIFIED.
533              THE ADD COMMAND IS  MEANINGFUL  ONLY  FOR  UNITS  THAT  WERE
534              PREVIOUSLY DROPPED.
535
536
537              6.3.6 DROP COMMAND
538
539              **************************************************************
540              DRO(P)/UNITS:<UNIT-LIST>
541              **************************************************************
542
543
544              6.3.6.1 UNITS SWITCH (/UNITS:<UNIT-LIST>)
545
546              <UNIT-LIST> IS AS IN THE RESTART COMMAND.
547
548
549              6.3.6.2 EFFECT OF DROP COMMAND
```

551
552
553
554
555
556   THE UNITS SPECIFIED WILL BE DROPPED FROM TESTING. THE UNITS
557   WILL BE RESELECTED ONLY BY THE EXECUTION OF AN ADD OR START
558   COMMAND.   THE UNITS SWITCH MUST BE ENTERED. THIS COMMAND
559   MUST BE FOLLOWED BY A RESTART OR A CONTINUE COMMAND.
560
561
562   6.3.7 PRINT COMMAND
563
564   ***
565   PRI(NT)
566   ***
567
568
569   6.3.7.1 EFFECT OF PRINT COMMAND
570
571   THE TOTAL NUMBER OF ERRORS FOR EACH UNIT SINCE THE LAST
572   START OR RESTART COMMAND ARE PRINTED. THE ISR (INHIBIT
573   STATISTICAL REPORTING) FLAG IS CLEARED.
574
575
576   6.3.8 DISPLAY COMMAND
577
578   ***
579   DIS(PLAY)/UNITS:<UNIT-LIST>
580   ***
581
582
583   6.3.8.1 UNITS SWITCH (/UNITS:<UNIT-LIST>)
584
585   <UNIT-LIST> IS AS IN THE RESTART COMMAND.
586
587
588   6.3.8.2 EFFECT OF DISPLAY COMMAND
589
590   THE HARDWARE P-TABLES FOR ALL UNITS UNDER TEST ARE PRINTED
591   OUT IN THE FORMAT IN WHICH THEY WERE ENTERED. ANY UNITS
592   THAT WERE DROPPED BY THE OPERATOR "DROP" COMMAND ARE SO
593   DESIGNATED.
594
595
596   6.3.9 FLAGS COMMAND
597
598   ***
599   FLA(GS)
600   ***
601
602
603   6.3.9.1 EFFECT OF FLAGS COMMAND
604
605   THE CURRENT SETTINGS OF ALL FLAGS ARE PRINTED.

```
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
```

6.3.10 ZFLAGS COMMAND

```
*****************************************************************
ZFL(AGS)
*****************************************************************
```

6.3.10.1 EFFECT OF ZFLAGS COMMAND

ALL FLAGS ARE CLEARED.

6.3.11 CONTROL CHARACTERS

A CONTROL C (C) ENTERED DURING THE EXECUTION OF A DIAGNOSTIC
CAUSES A RETURN TO COMMAND MODE.

A CONTROL Z (Z) ENTERED DURING ONE OF THE THREE OPERATOR
DIALOGUES- INITIAL DIALOGUE (SEE 6.2),HARDWARE DIALOGUE (SEE
6.3.1.5), OR SOFTWARE DIALOGUE (SEE 6.3.1.5) CAUSES THE
DEFAULTS TO BE TAKEN FOR THE REMAINDER OF THAT DIALOGUE.

A CONTROL O (O) ENTERED DURING THE EXECUTION OF A DIAGNOSTIC
CAUSES ALL TELETYPE OUTPUT TO BE SURPRESSED FOR   THE
REMAINDER OF THE DIAGNOSTIC OR UNTIL ANOTHER O IS TYPED,
WHICH RESTORES NORMAL TELETYPE OUTPUT.

6.3.12 HARDWARE PARAMETERS

THE FOLLOWING QUESTIONS WILL BE ASKED ON A START COMMAND.
THE VALUE LOCATED TO THE LEFT OF THE QUESTION MARK IS THE
DEFAULT VALUE THAT WILL BE TAKEN ON A CARRIAGE RETURN
RESPONSE.

2. MICRO-CPU CSR ADDRESS: (O) 177000?

THIS IS THE ADDRESS AT WHICH THE CSR REGISTERS (SELO) RESIDE
ON THE UNIBUS.   THE ALLOWABLE RANGE IS   160000-177776
(OCTAL), AND THE DEFAULT IS 177000.

3. MICRO CPU VECTOR ADDRESS: (O) 300?

THE ALLOWABLE RANGE IS 300-770,AND DEFAULT VALUE IS 300

660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715

4. MICRO CPU PRIORITY LEVEL: (4) 7?

DEFFAULT VALVE IS 4
NOTE:
M7500 AND M7501 MODULE MOUNTED WITH DC003 CHIPS CAN ONLY
INTERUPT ON LEVEL 4

6.3.13 SOFTWARE PARAMETERS

NO SOFTWARE PARAMETER QUESTIONS ARE ASKED IN THAT
STATIC LOGIC TESTS.

6.3.14 EXTENDED DISCUSSION OF P-TABLE DIALOGUE

THE FULL CAPABILITY OF THE HARDWARE DIALOGUE IS REVEALED BY
THE FOLLOWING DISCUSSION OF WHAT HAPPENS INTERNALLY.

AS SOON AS THE QUESTION "# UNITS?" IS ANSWERED (WITH THE
NUMBER N, SAY) SPACE IN CORE IS ALLOCATED FOR N P-TABLES.
ALL OF THE P-TABLES ARE OF THE SAME FORMAT, AND THERE IS A
ONE-TO ONE CORRESPONDENCE BETWEEN THE HARDWARE PARAMETER
QUESTIONS AND THE SLOTS IN THE P-TABLE FORMAT.

ON THE FIRST TRIP THRU THE QUESTIONS, ALL OF THE SLOTS IN
ALL OF THE P-TABLES ARE FILLED. IF THE OPERATOR TYPES IN
LESS THAN N EXPLICIT VALUES IN RESPONSE TO A PARTICULAR
QUESTION, THESE VALUES ARE PLACED IN THE P-TABLES (ONE VALUE
GOING INTO THE PROPER SLOT OF EACH P-TABLE BEGINNING WITH
THE FIRST P-TABLE) UNTIL THE STRING OF VALUES IS EXHAUSTED.
THE LAST VALUE IN THE STRING BECOMES THE NEW DEFAULT AND IS
USED TO FILL THAT SLOT IN THE REMAINING P-TABLES.

ON SUBSEQUENT TRIPS THRU THE QUESTIONS, THE SAME PROCESS IS
CARRIED OUT, EXCEPT THAT THE EARLIEST P-TABLE NOT TO HAVE
RECEIVED AN EXPLICIT VALUE IN ANY OF ITS SLOTS NOW ASSUMES
THE ROLE THAT TABLE NUMBER ONE PLAYED IN THE FIRST TRIP.

THE SERIES OF QUESTIONS IS REISSUED UNTIL AT LEAST ONE
QUESTION HAS RECEIVED N EXPLICIT VALUES FROM THE OPERATOR.

IN GIVING A STRING OF VALUES, COMMAS WITHOUT INTERVENING
VALUES MAY BE USED TO INDICATE A REPETITION OF THE LAST
NAMED VALUE.

A STRING OF VALUES MAY BE GIVEN AS A RANGE (6-10 FOR
EXAMPLE). IF THE VALUES REPRESENT PURE NUMERICAL DATA, THIS
SAMPLE RANGE TRANSLATES TO THE STRING 6,7,8,9,10 (AN
INCREMENT OF 1). IF THE VALUES ARE ADDRESSES, THE SAMPLE
RANGE TRANSLATES TO THE STRING 6,8,10 (AN INCREMENT OF 2).

```
717
718
719
720
721
722          NOW  LET  US  SEE  HOW  WE  COULD  USE THESE CAPABILITIES TO
723          CONSTRUCT A SET OF P-TABLES.  ASSUME THAT WE HAVE 16  UNITS,
724          AND THAT THERE ARE THREE HARDWARE PARAMETERS FOR EACH (THREE
725          SLOTS  IN  THE  P-TABLE,  THREE  HARDWARE  QUESTIONS  IN  THE
726          DIALOGUE).  LET THE DESIRED VALUE FOR THE FIRST PARAMETER BE
727          THE NUMBER 75 FOR ALL 16 TABLES.  LET THE DESIRED VALUE  FOR
728          THE   SECOND   PARAMETER   BE   EQUAL  TO  THE  UNIT  NUMBER
729          (0,1,2,....,15) EXCEPT FOR UNIT 12, WHICH SHOULD RECEIVE  THE
730          VALUE  11.  LET THE DESIRED VALUE FOR THE THIRD PARAMETER BE
731          THE NUMBER 76 FOR THE FIRST 7 UNITS AND THE  NUMBER  77  FOR
732          THE LAST 9 UNITS.
733
734          THE FOLLOWING DIALOGUE WOULD ACCOMPLISH THIS GOAL:
735
736          # UNITS (D) ? 16
737
738          UNIT 1
739          <QUESTION 1> ? 75
740          <QUESTION 2> ? 0-6
741          <QUESTION 3> ? 76
742
743          UNIT 21
744          <QUESTION 1> ?
745          <QUESTION 2> ? 7-11,,13-15
746          <QUESTION 3> ? 77
747
748          THE  FIRST  TIME THE SERIES IS ASKED, SLOT ONE RECEIVES A 75
749          IN ALL 16 TABLES.  SLOT TWO RECEIVES THE VALUES  0,1,2,....,6
750          IN  TABLES  0  THRU  6 AND A CONSTANT 6 IN TABLES 7 THRU 15.
751          SLOT THREE RECEIVES A CONSTANT 76 IN ALL 16 TABLES.
752
753          THE SECOND TIME THRU THE SERIES, TABLES 16 THRU THE END  ARE
754          GOING TO BE AFFECTED (NOTE THAT THIS PIECE OF INFORMATION IS
755          PRINTED OUT FOR THE THE OPERATOR IN THE FORM 'UNIT  XX'  AT
756          THE  BEGINNING  OF EACH SERIES).  QUESTION 1 IS RESPONDED TO
757          BY A <CR>, SO SLOT ONE STAYS AT CONSTANT 75 IN TABLES 7 THRU
758          15, SINCE NO NEW EXLICIT VALUES ARE TYPED IN.  SLOT TWO GETS
759          THE VALUES 7,8,9,10,11 IN TABLES 7 THRU 11, AND GETS A 11 IN
760          SLOT  12, AND GETS THE VALUES 13,14,15 IN TABLES 13 THRU 15.
761          SLOT THREE GETS THE VALUE 77 IN TABLES 7 THRU 15.
762
763          THE DIALOGUE IS TERMINATED WHEN THE SOFTWARE RECOGNIZES THAT
764          16 EXPLICIT VALUES HAVE BEEN GIVEN FOR AT LEAST ONE QUESTION
765          (NAMELY QUESTION 2).
```

```
767
768
769
770
771
772             7.0  TEST DESCRIPTIONS
773
774
775             ******************** TEST 1 ********************
776             *VERIFY THAT REFERENCED QBUS DEVICE REGISTERS
777             *DO NOT CAUSE TIME OUT TRAP
778             ************************************************
779
780
781
782             ******************** TEST 2 ********************
783             *
784             *CLEAR ALL KMV11 REGISTERS AND CHECK
785             *
786             ************************************************
787
788
789
790             ******************** TEST 3 ********************
791             *
792             *CHECK QBUS ACCESS ON KMV11 REGISTERS (FROM SEL2 TO SEL16)
793             *
794             ************************************************
795
796
797
798             ******************** TEST 4 ********************
799             *
800             *CHECK Q BUS ACCESS ON REGISTER SEL0
801             *
802             ************************************************
803
804
805             ******************** TEST 5 ********************
806             *
807             *CHECK Q BUS BYTE ACCESS ON  ALL KMV11 REGISTERS
808             *
809             ************************************************
810
811
812             ******************** TEST 6 ********************
813             *
814             *DATA TRANSFER TO REGISTER SEL 2
815             *
816             ************************************************
817
818
819             ******************** TEST 7 ********************
820             *
821             *DATA TRANSFER TO REGISTER SEL 4
822             *
823             ************************************************
```

```
824
825
826            ********************** TEST 8 *******************
827            *
828            *DATA TRANSFER TO REGISTER SEL 6
829            *
830            ***************************************************
831
832
833            ********************** TEST 9 *******************
834            *
835            *DATA TRANSFER TO REGISTER SEL 10
836            *
837            ***************************************************
838
839
840            ********************** TEST 10 *******************
841            *
842            *DATA TRANSFER TO REGISTER SEL 12
843            *
844            ***************************************************
845
846
847            ********************** TEST 11 *******************
848            *
849            *DATA TRANSFER TO REGISTER SEL 14
850            *
851            ***************************************************
852
853
854            ********************** TEST 12 *******************
855            *
856            *DATA TRANSFER TO REGISTER SEL 16
857            *
858            ***************************************************
859
860
861            ********************** TEST 13 *******************
862            *
863            *CHECK DATA TRASFERS USING ALL CSR'S REGISTERS
864            *
865            ***************************************************
866
867
868            ********************** TEST 14 *******************
869            *
870            *KMV11 RAM MEMORY TEST: MEMORY PATERN TEST
871            *
872            ***************************************************
873
874
875            ********************** TEST 15 *******************
876            *
877            *KMV11 RAM MEMORY TEST: MEMORY ADDRESS TEST
878            *
879            ***************************************************
880
```

```
881
882                          ********************** TEST 16 *******************
883                          *
884                          *KMV11 RAM MEMORY TEST: MEMORY ADDRESS COMPLEMENT TEST
885                          *
886                          ***************************************************
887
888
889                          ********************** TEST 17 *******************
890                          *
891                          *CHECK PROM REVISION
892                          *
893                          ***************************************************
894
895
896                          ********************** TEST 18 *******************
897                          *
898                          *PROM CHECKSUM TEST
899                          *
900                          ***************************************************
901
902
903                          ********************** TEST 19 *******************
904                          *
905                          *DMA TRANSFER INTO KMV11
906                          *
907                          ***************************************************
908
909
910                          ********************** TEST 20 *******************
911                          *
912                          *TEST DMA TRANSFERS OUT KMV11
913                          *
914                          ***************************************************
915
916
917                          ********************** TEST 21 *******************
918                          *
919                          *TEST DMA TRANSFERS IN BOTH DIRECTION
920                          *
921                          ***************************************************
922
923
924                          ********************** TEST 22 *******************
925                          *
926                          *TEST INTERUPT CAPABILITY OF KMV11 MODULE ON QBUS
927                          *
928                          ***************************************************
929
930
931                          ********************** TEST 23 *******************
932                          *
933                          *TEST INTERUPT ON DCT11 MICROPROCESSOR
934                          *
935                          ***************************************************
```

937
938
939   8.0 ERROR INFORMATION
940
941
942   8.1 ERROR REPORTING
943
944   ERRORS ARE REPORTED BY THE PROGRAM AS THEY OCCUR (IF NOT
945   INHIBITED). THE REPORT CONFORMS TO THE DIAGNOSTIC SUPERVISOR
946   ERROR REPORT FORMAT, AND CONSISTS OF A DESCRIPTION OF THE
947   ERROR, THE TEST NUMBER, SUBTEST NUMBER, PC OF THE ERROR
948   CALL, DEVICE ADDRESS, AND BASIC AND EXTENDED ERROR
949   INFORMATION.
950
951
952
953   THE FOLLOWING EXAMPLES PROVIDE TYPICAL ERROR REPORTS:
954
955
956   ;CZDMQ DVC FTL ERR 00045 TST 027 SUB 000 PC:022572
957
958   ;MASTER   CLEAR FAILED TO CLEAR PC REG, CONTENTS=000624
959   ;CZDMQ DVC FTL ERR 00015 TST 042 SUB 000 PC:027234
960
961   ;UNIT=00, FAILING UNIT ADDRESS=160170
962   ;JUMP TEST ERROR
963   ;FROM ADDR   TO ADDR   BAD ADDR
964   ;000402   000000   000114
965
966
967
968   FOR ALL OTHER ERRORS, THE REPORT MAY BE MORE EXTENSIVE AND
969   REQUIRE ADDITIONAL DATA TO BE REPORTED.
970
971
972
973   9.0   HISTORY
974
975   -   DESIGN STARTED ON JANUARY 82
976   -   REVIEW ON DECEMBER 82
977
978   a
979
980
981
982
983

```
 985                                    .TITLE KMV11 A/B LOGIC DIAG
 993          002000                        .=2000
 994
 995
 996
 997
 998
 999
1000                                    .MCALL  SVC
1001 002000                             SVC                             ; INITIALIZE SUPERVISOR MACROS
1002
1003
1004
1005
1006
1007 002000                             BGNMOD  KMV11A.B
1008
1009
1010          000000                    $LSTIN= 0
1011          000000                    $LSTTAG= 0
1012          177777                    SVCINS= -1       ; LIST INSTRUCTIONS, SHIFTED RIGHT
1013          177777                    SVCTST= -1       ; LIST TEST TAGS, SHIFTED RIGHT
1014          177777                    SVCSUB= -1       ; LIST SUBTEST TAGS, SHIFTED RIGHT
1015          177777                    SVCGBL= -1       ; LIST GLOBAL TAGS, SHIFTED RIGHT
1016          177777                    SVCTAG= -1       ; LIST OTHER TAGS, SHIFTED RIGHT
1017
1018                                    ;      CHANGE THE VALUES OF THE SVC... SYMBOLS TO BE ZERO IF YOU WISH
1019                                    ;      TO ALIGN THE MACRO CALLS AND THEIR EXPANSIONS.  CHANGE THE
1020                                    ;      SYMBOLS TO BE MINUS-ONE TO NOT LIST THE EXPANSIONS.  YOU MAY
1021                                    ;      CHANGE THE SYMBOLS AT ANY POINT IN YOUR PROGRAM.
1022
1023
```

```
KMV11 A/B LOGIC DIAG    MACRO M1200  06-JAN-83 09:39  PAGE 17
PROGRAM HEADER

 1025                                    .SBTTL  PROGRAM HEADER
 1026                                    ;++
 1027                                    ; THE PROGRAM HEADER IS THE INTERFACE BETWEEN
 1028                                    ; THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
 1029                                    ;--
 1030
 1031 002000                                    POINTER BGNSW,BGNDU,BGNSETUP
 1032
 1033
 1034
 1052
 1053 002000                                    HEADER  VKMAA0,A,0,240.,0
 1054
 1055
 1066
 1067
 1068
 1069
 1070                 ;%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 1071
 1072
 1073
 1074
 1075
 1076                                    ;++
 1077                                    ; THIS TABLE IS USED BY THE RUNTIME SERVICES
 1078                                    ; TO PROTECT THE LOAD MEDIA.
 1079                                    ;--
 1080
 1081 002122                                    BGNPROT
 1082
 1083 002122  000000                            0              ;OFFSET INTO P-TABLE FOR CSR ADDRESS
 1084 002124  177777                            -1             ;OFFSET INTO P-TABLE FOR MASSBUS ADDRESS
 1085 002126  177777                            -1             ;OFFSET INTO P-TABLE FOR DRIVE NUMBER
 1086
 1087
 1101
 1102
 1103 002130                                    ENDPROT
 1104
```

```
1106                                    .SBTTL  DISPATCH TABLE
1107
1108                        ;////////////////////////////////////////////////////////////////////
1109                        ;/ THE DISPATCH TABLE CONTAINS THE STARTING ADDRESS OF EACH TEST.
1110                        ;/ IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.
1111                        ;////////////////////////////////////////////////////////////////////
1112
1113 002130                        DISPATCH 23
1114
1121
1122
1123
1124
1125
1126
1127
1128                        ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
1129
1130
1131
1132
1133
1134
1135
1136
1137                        .SBTTL DEFAULT HARDWARE P-TABLE
1138
1139                        ;////////////////////////////////////////////////////////////////////////
1140                        ;/ THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF
1141                        ;/ THE TEST-DEVICE PARAMETERS.  THE STRUCTURE OF THIS TABLE
1142                        ;/ IS IDENTICAL TO THE STRUCTURE OF THE RUN-TIME P-TABLE.
1143                        ;/ AND IS USED AS A '' TEMPLATE'' FOR BUILDING THE P-TABLE
1144                        ;////////////////////////////////////////////////////////////////////////
1145
1146                        .ENABL  AMA
1147 002210                        BGNHW   DFPTBL
1148
1158
1159
1160 002212  177000                .WORD   177000          ;KMV11.CSRS ADDRESS
1161 002214  000300                .WORD   300             ;KMV11, VECTOR ADDRESS
1162 002216  004000                .WORD   4000            ;INTERRUPT PRIORITY LEVEL (4)
1163 002220  000001                .WORD   1
1164
1165 002222                        ENDHW
```

```
1167
1168
1169
1170                              .SBTTL  GLOBAL EQUATES SECTION
1171
1172
1173                              ;////////////////////////////////////////////////////////////////////////////
1174                              ;/      THE GLOBAL EQUATES SECTION CONTAINS PROGRAM EQUATES THAT
1175                              ;/      ARE USED IN MORE THAN ONE TEST.
1176                              ;////////////////////////////////////////////////////////////////////////////
1177
1178
1179
1189
1190
1205
1206 002222                                    EQUALS
                                 ;
                                 ; BIT DIFINITIONS
                                 ;
         100000                  BIT15== 100000
         040000                  BIT14== 40000
         020000                  BIT13== 20000
         010000                  BIT12== 10000
         004000                  BIT11== 4000
         002000                  BIT10== 2000
         001000                  BIT09== 1000
         000400                  BIT08== 400
         000200                  BIT07== 200
         000100                  BIT06== 100
         000040                  BIT05== 40
         000020                  BIT04== 20
         000010                  BIT03== 10
         000004                  BIT02== 4
         000002                  BIT01== 2
         000001                  BIT00== 1

         001000                  BIT9==  BIT09
         000400                  BIT8==  BIT08
         000200                  BIT7==  BIT07
         000100                  BIT6==  BIT06
         000040                  BIT5==  BIT05
         000020                  BIT4==  BIT04
         000010                  BIT3==  BIT03
         000004                  BIT2==  BIT02
         000002                  BIT1==  BIT01
         000001                  BIT0==  BIT00
                                 ;
                                 ; EVENT FLAG DEFINITIONS
                                 ;    EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
                                 ;
         000040                  EF.START==      32.                      ; START COMMAND WAS ISSUED
         000037                  EF.RESTART==    31.                      ; RESTART COMMAND WAS ISSUED
         000036                  EF.CONTINUE==   30.                      ; CONTINUE COMMAND WAS ISSUED
         000035                  EF.NEW==        29.                      ; A NEW PASS HAS BEEN STARTED
         000034                  EF.PWR==        28.                      ; A POWER-FAIL/POWER-UP OCCURRED
                                 ;
```

```
                                ;
                                ; PRIORITY LEVEL DEFINITIONS
                                ;
              000340            PRI07== 340
              000300            PRI06== 300
              000240            PRI05== 240
              000200            PRI04== 200
              000140            PRI03== 140
              000100            PRI02== 100
              000040            PRI01= 40
              000000            PRI00== 0
                                ;
                                ;OPERATOR FLAG BITS
                                ;
              000004            EVL==        4
              000010            LOT==       10
              000020            ADR==       20
              000040            IDU==       40
              000100            ISR==      100
              000200            UAM==      200
              000400            BOE==      400
              001000            PNT==     1000
              002000            PRI==     2000
              004000            IXE==     4000
              010000            IBE==    10000
              020000            IER==    20000
              040000            LOE==    40000
              100000            HOE==   100000
1207
1208
1209          000340            MAXPRI==340
1210
1211          054000            MAINT0==54000        ;MASTER CLEAR = 1,MODE = 1 ,MAINT 1 = 1 ,T11=HOLD
1212          044000            MAINT1==44000        ;MASTER CLEAR = 1,MODE = 0 ,MAINT 1 = 0 ,T11=NOT HOLD
1213          040000            MCLR==40000
1214          052525            DATA1== 052525
1215          125252            DATA2== 125252
1216
1217
1218                            ;;*****************************************************************************
1219                            ;* PROGRAM EVENT FLAG DEFINITIONS
1220                            ;;*****************************************************************************
1221
1222
1223
```

```
1225                                    .SBTTL  GLOBAL DATA SECTION
1226
1227                            ;////////////////////////////////////////////////////////////////////////
1228                            ;/       THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
1229                            ;/       IN MORE THAN ONE TEST.
1230                            ;////////////////////////////////////////////////////////////////////////
1231
1237
1238
1239                            ;;*******************************************************************
1240                            ;* STORAGE FOR DEVICE REGISTERS
1241                            ;;*******************************************************************
1242 002222                             DESCRIPT          <KMV11A.B  LOGIC DIAGNOSTIC>
1243
1244
1245
1258
1259 002256                             ERRTBL
     002256  000000          ERRTYP::           .WORD   0
     002260  000000          ERRNBR::           .WORD   0
     002262  000000          ERRMSG::           .WORD   0
     002264  000000          ERRBLK::           .WORD   0
1260
1261
1262
1263
1264                            ;*******************************************************************
1265                            ;* PROGRAM CONTROL PARAMETERS
1266                            ;*******************************************************************
1267 002266  000000          L$SW:   .WORD   0
1268 002270  000015          L$UIT:  .WORD   15
1269 002272  000000          UNIT:   .WORD   0
1270 002274  000000          LOCK:   .WORD   0       ;ADDRESS FOR LOCK CURRENT DATA
1271 002276  000005          MAXERR: .WORD   5       ;MAX ERROR CNT BEFORE DROPPING UNIT
1272 002300  000000          ERRCNT: .WORD   0       ;ERROR CNT
1273 002302  000000          LOGDEV: .WORD   0       ;LOGICAL DEVICE NUMBER
1274 002304  000000          PSTACK: .WORD   0       ;BASE LEVEL PROGRAM STACK POINTER
1275 002306  000000          SAVSP:  .WORD   0       ;STACK POINTER STORAGE
1276 002310  000000          SAVPC:  .WORD   0       ;PROGRAM COUNTER STORAGE
1277 002312  000000          SAVE4:  .WORD   0
1278 002314  000000          SAVE6:  .WORD   0
1279 002316  000000          FTIME:  .WORD   0
```

```
KMV11 A/B LOGIC DIAG    MACRO M1200  06-JAN-83 09:39  PAGE 21
GLOBAL DATA SECTION

1281                                ;******************************************************************
1282                                ;* MISCELLANEOUS STORAGE
1283                                ;******************************************************************
1284 002320  000000                 FLAG:    .WORD   0
1285 002322  000000                 DH1:     .WORD   0
1286 002324  000000                 DELCT1:  .WORD   0
1287 002326  000000                 DELCT2:  .WORD   0
1288 002330  000000                 GOOD:    .WORD   0
1289 002332  000000                 GOOD0:   .WORD   0
1290 002334  000000                 GOOD1:   .WORD   0
1291 002336  000000                 GOOD2:   .WORD   0
1292 002340  000000                 GOOD4:   .WORD   0
1293 002342  000000                 GOOD6:   .WORD   0
1294 002344  000000                 GOOD10:  .WORD   0
1295 002346  000000                 GOOD12:  .WORD   0
1296 002350  000000                 GOOD14:  .WORD   0
1297 002352  000000                 GOOD16:  .WORD   0
1298 002354  000000                 SEL0:    .WORD   0
1299 002356  000000                 SEL1:    .WORD   0
1300 002360  000000                 SEL2:    .WORD   0
1301 002362  000000                 SEL4:    .WORD   0
1302 002364  000000                 SEL6:    .WORD   0
1303 002366  000000                 SEL10:   .WORD   0
1304 002370  000000                 SEL12:   .WORD   0
1305 002372  000000                 SEL14:   .WORD   0
1306 002374  000000                 SEL16:   .WORD   0
1307 002376  000000                 BSEL1:   .WORD   0
1308 002400  000000                 RANST:   .WORD   0
1309 002402  000000                 RANSEL:  .WORD   0
1310 002404  000000                 RANMTA:  .WORD   0
1311 002406  000000                 RANDN:   .WORD   0
1312 002410  000000                 SAVPC1:  .WORD   0
1313 002412  000000                 SAVSTA:  .WORD   0
1314 002414  000000                 COUNT:   .WORD   0
1315 002416  000000                 NUMBER:  .WORD   0
1316 002420  000000                 ADDR:    .WORD   0
1317 002422  000000                 GDDAT:   .WORD   0
1318 002424  000000                 BDDAT:   .WORD   0
1319
1320 002426                         TTABLE:  .BLKW   2000
1321 006426                         RTABLE:  .BLKW   2000
1322
1323 012426  000000                 EXADDR:  .WORD   0
1324 012430  000000                 INTFLG:  .WORD   0
1325 012432  000000                 BAD:     .WORD   0
1326 012434  000000                 BSEL0:   .WORD   0
1327 012436  000000                 DATA:    .WORD   0
1328 012440  000000                 VECT:    .WORD   0
1329
1330 012442  000000                 KIND:    .WORD   0        ;=0 IF KMV11A ,=1 IF KMV11B
1331 012444  000000                 CHANEL:  .WORD   0
1332
1333 012446  000000                 TXDATA:  .WORD   0
1334 012450  000000                 RXDATA:  .WORD   0
1335 012452  000000                 TSPEED:  .WORD   0
1336 012454  000000                 LENGTH:  .WORD   0
1337 012456  000000                 NUB:     .WORD   0
```

C 3

KMV11 A/B LOGIC DIAG   MACRO M1200 06-JAN-83 09:39 PAGE 21-1
GLOBAL DATA SECTION

```
1338 012460 000000                RXCNT:  .WORD  0
1339 012462 000000                MAXCNT: .WORD  0
1340
```

```
1342
1343                                 ;*******************************************************************
1344                                 ;LOAD IN LOCATION ''GDREV'' THE PROM VERSION NUMBER THAT IS        *
1345                                 ;COMPATIBLE WhITH THIS DIAGNOSTIC                                 *
1346                                 ;                                                                 *
1347                                 ;EACH PROM CONTAIN A REV LEVEL AND A ECO LEVEL:                   *
1348                                 ;THE REV LEVEL IS MODIFIED EACH TIME A MODIFICATION IS DONE       *
1349                                 ;THE ECO LEVEL IS MODIFIED WHEN THE PROM MODIFICATION NEED        *
1350                                 ;A DIAGNOSTIC MODIFICATION                                        *
1351                                 ;*******************************************************************
1352
1353
1354
1355
1356
1357
1358 012464  000001                  GDREV:  .WORD   1
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
```

```
1370                              ;**********************************************************************
1371                              ;* PROGRAM CONTROL FLAGS
1372                              ;**********************************************************************
1373 012466    000      INIFLG: .BYTE   0                       ;PROGRAM INITIALIZING FLAG
1374                              .EVEN
1375 012470    000      LOKFLG: .BYTE   0                       ;LOCK ON CURRENT TEST FLAG
1376 012471    000      QV.FLG: .BYTE   0                       ;QUICK VERIFY FLAG
1377                              .EVEN
1378 012472  000000     UUT:    .WORD   0                       ;CURRENT UNIT UNDER TEST
1379
1380
1381                              ;**********************************************************************
1382                              ;* POINTERS TO KMV11 VECTORS AND REGISTERS
1383                              ;**********************************************************************
1384 012474  000000     KMVV00: 0            ;POINTER TO KMV11  INTRPT VECTOR 0
1385 012476  000000     KMVLVL: 0            ;POINTER TO KMV11  INTRPT SERVICE
1386 012500  000000     KMVV04: 0            ;POINTER TO KMV11  INTRPT VECTOR 04
1387 012502  000000     KMVV02: 0            ;         ''        ''         ''      ''     02
1388 012504  000000     KMVV06: 0            ;         ''        ''         ''      :      06
1389 012506  000000     KMTLVL: 0            ;POINTER TO KMV11 TX INTRPT SERVICE PS
1390 012510  000000     KMVCSR: 0            ;POINTER TO KMV11 CONTROL STATUS REGISTER
1391 012512  000000     KMVP02: 0            ;POINTER TO KMV11 PORT REGISTER - SEL2
1392 012514  000000     KMVP04: 0            ;POINTER TO KMV11 PORT REGISTER - SEL4
1393 012516  000000     KMVP06: 0            ;POINTER TO KMV11 PORT REGISTER - SEL6
1394
1395 012520  000000     KMVP10: 0            ;POINTER TO KMV11 PORT REG  -SEL10
1396 012522  000000     KMVP12: 0            ;POINTER TO PORT REG  -SEL 14
1397 012524  000000     KMVP14: 0            ;POINTER TO PORT REG  -SEL14
1398 012526  000000     KMVP16: 0            ;POINTER TO PORT REG 16
1399
1400 012530  000000     LOOP:   0            ;POINTER TO LOOP BACK CONNECTOR
1401
1402
1403                              ;;****  PRIMARY REG ADRS STORAGE FOR THIS UNIT  ******
1404                              ;THESE LOCATIONS WILL BE LOADED FOR THE CURRENT UNIT, IN INIT CODE
1405 012532             REGADR:
1406
1407                              ;;****  STACK USED FOR SUBROUTINE LINKAGE  ****
1408 012532                      .BLKW   100
1409 012732             SSTACK:
1410
1411
1412
1413
1414
1415
1416
```

```
1418                            .SBTTL  GLOBAL TEXT SECTION
1419
1420            ;%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1421            ;%      THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
1422            ;%      MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
1423            ;%      MORE THAN ONE TEST.
1424            ;%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1425
1426            ;;**********************************************************************
1427            ;* NAMES OF DEVICES SUPPORTED BY PROGRAM
1428            ;;**********************************************************************
1429 012732            DEVTYP  <M7500 OR M7501 MODULE>
1430
1431
1432            ;
1433            ; FORMAT STATEMENTS USED IN PRINT CALLS
1434            ;
1435
1442
1443
1444
1445
1446
```

```
  1448                               .SBTTL  GLOBAL SUBROUTINES
  1449
  1450
  1451
  1452
  1453
  1454                      ;-------------------------------------------------------------
  1455                      ; MACRO'S NEEDED TO CALL SUBROUTINES
  1456                      ;-------------------------------------------------------------
  1457
  1458                      .MACRO  CLRMAR
  1459                              ROMCLK
  1460                              004000
  1461                      .ENDM   CLRMAR
```

KMV11 A/B LOGIC DIAG   MACRO M1200   06-JAN-83 09:39 PAGE 26
GLOBAL SUBROUTINES

```
1463                                  ;ROUTINE TO WAIT FOR EVENT OR TIMEOUT
1464
1465
1466
1467                                  ;CALLING SEQUENCE:      JSR   PC,WAIT1
1468                                  ;                       JSR   PC,WAIT2
1469
1470
1471                                  ;INPUTS PARAMETERS:     DELCT1,DELCT2
1472
1473
1474                                  ;                       INC DELCT1 UNTIL 0
1475                                  ;                       DEC DELCT2 UNTIL 0     DELCT2= NUMB OF WAIT1 PASSES
1476
1477
1478
1479
1480
1481
1482
1483 012760  005237  002324          WAIT2:  INC     DELCT1
1484 012764  001375                          BNE     WAIT2
1485
1486 012766                                  BREAK
1487
1488 012770  005337  002326                  DEC     DELCT2
1489 012774  001371                          BNE     WAIT2
1490
1491 012776  000207                          RTS     PC
1492
1493
1494
1495
1496
1497
1498 013000  005237  002324          WAIT1:  INC     DELCT1
1499 013004  001375                          BNE     WAIT1
1500
1501 013006  000207                          RTS     PC
```

```
1503                          ;MACRO TO WAIT A FEW MS
1504
1505
1506                          ;CALLING SEQUENCE:     WAITA   X               0<X<177777
1507                          ;                      WAITB   X,Y             0<X OR Y<177777
1508
1509
1510
1511                          .MACRO  WAITA   X
1512                                  MOV     #X,DELCT1               ;LOAD COUNT
1513                                  JSR     PC,WAIT1                ;WAIT
1514                          .ENDM
1515
1516
151/
1518
1519
1520
1521
1522                          .MACRO  WAITB   X,Y
1523                                  MOV     #X,DELCT1
1524                                  MOV     #Y,DELCT2
1525                                  JSR     PC,WAIT2
1526                          .ENDM
1527
1528
1529
1530
```

```
1532                                    ;ROUTINE TO DROP UNIT AFTER 5 ERROR
1533
1534
1535                                    ;JSR   PC,CHKMAX
1536
1537
1538
1539
1540
1541
1542
1543
1544  013010                   CHKMAX:  INLOOP                        ;LOOPING ON ERROR?
1545  013012                            BCOMPLETE        1$           ;IF YES, EXIT
1546
1547
1548  013014                            RFLAGS   R0                   ;GET OPERATOR FLAG
1549  013016   032700   000040          BIT      #IDU,R0             ;IS DROPPING INHIBITTED?
1550  013022   001026                   BNE      1$                  ;IF YES EXIT
1551
1552
1553  013024   005237   002300          INC      ERRCNT              ;UPDATE ERROR COUNT
1554  013030   023737   002300  002276  CMP      ERRCNT,MAXERR       ;TOO MANY ERROR?
1555  013036   003420                   BLE      1$                  ;IF NOT JUMP
1556
1557
1558  013040                            PRINTF   #NERRS,MAXERR,UUT   ;TOO MANY ERROR!
1559  013070                            DODU     UUT                 ;DROP UNIT
1560
1561  013076                            DOCLN                        ;END THE SUBPASS
1562
1563  013100   000207           1$:     RTS      PC
1564
1565
1566
1567
1568
1569  013102    045    116    045 NERRS: .NLIST  BEX
1570                                    .ASCIZ  /%N%AMORE THAN %D3%A  ERRORS ON UNIT %D2/
1571                                    .LIST   BEX
1572                                    .EVEN
1573
1574
```

```
1576                                 ;ROUTINE TO CHECK REGISTER BSEL0 AND TO REPORT ERROR
1577
1578
1579
1580
1581
1582
1583                                 ;CALLING SEQUENCE:      JSR     PC,TSTERR
1584
1585
1586
1587                                 ;OUTPUT PARAMETERS:     RETURN TO       PC      IF TEST IS OK
1588                                 ;                                       PC+2    IF TIMEOUT DURING TEST
1589                                 ;                                       PC+4    IF NO KMV11 ANSWER
1590                                 ;                                       PC+6    IF DATA CMP ERROR
1591
1592
1593
1594
1595
1596
1597 013152  004537  013722  TSTERR: JSR     R5,CBSEL0               ;LOOK IF BSEL0=0
1598 013156  000000                  .WORD   0
1599 013160  000411                  BR      1$                      ;TEST IS OK ,RTS PC
1600
1601
1602 013162  122737  000200  012434  CMPB    #200,BSEL0              ;LOOK IF BSEL0=200
1603 013170  001406                  BEQ     2$                      ;TIMEOUT DURING TEST,RTS PC+2
1604
1605
1606 013172  122737  000100  012434  CMPB    #100,BSEL0              ;LOOK IF BSEL0=100
1607 013200  001405                  BEQ     3$                      ;DATA CMP ERROR,RTS PC+6
1608
1609
1610
1611 013202  000407                  BR      4$                      ;NO KMV11 ANSWER ,RTS PC+4
1612
1613
1614
1615 013204  000207          1$:     RTS     PC                      ;TEST OK
1616
1617
1618 013206  062716  000002  2$:     ADD     #2,(SP)
1619 013212  000207                  RTS     PC                      ;TIMEOUT ERROR
1620
1621
1622 013214  062716  000006  3$:     ADD     #6,(SP)
1623 013220  000207                  RTS     PC                      ;DATA CMP ERROR
1624
1625
1626 013222  062716  000004  4$:     ADD     #4,(SP)
1627 013226  000207                  RTS     PC                      ;NO KMV11 ANSWER
```

```
1629                                        .SBTTL   NUMBER GENERATOR
1630
1631
1632
1633                                        DESCRIPTION:
1634
1635                                            ROUTINE TO GENERATE DATA PATTERNS,
1636                                        THE TYPE OF PATTERN IS SELECTED BY R3, AND THE
1637                                        PATTERN GENERATED IS RETURNED IN LOCATION "DATA"
1638                                        AND LOCATION "GOOD"
1639
1640                                        CALLING SEQUENCE:
1641
1642                                            JSR    PC,GENER
1643
1644                                        INPUT PARAMETERS:
1645
1646                                        R3 CONTAINS THE PATTERN NUMBER
1647
1648                                        R3=0              ALL ZEROES
1649         1                ALL ONES
1650         2                010101 ETC BIT PATTERN
1651         3                101010 ETC BIT PATTERN
1652         4                ROTATING 1 IN A ZERO WORD
1653         5                ROTATING 0 IN AN ALL ONE WORD
1654         6                PSEUDO RANDOM NUMBER
1655         7                INCREMENTING DATA PATTERN, GOOD
1656                          CONTAINS THE VALUE TO BE UPDATED
1657
1658
1659
1660
1661                                        IMPLICIT INPUT PARAMETERS:
1662
1663                                            NONE
1664
1665
1666                                        OUTPUT PARAMETERS:
1667
1668                                            THE NUMBER GENERATED IS HELD IN
1669                                        DATA AND GOOD.
1670
1671
1672                                        IMPLICIT OUTPUT PARAMETERS:
1673
1674                                            NONE
1675
1676
1677                                        COMPLETION CODES:
1678
1679                                            NONE
1680
1681
1682                                        POSSIBLE ERROR CODES:
1683
1684                                            NONE
1685
```

```
1686                                        ;
1687                                        ;
1688 013230  042703  177770    GENER:  BIC     #177770,R3
1689 013234  004737  013530            JSR     PC,SAVREG
1690 013240  006303                    ASL     R3
1691 013242  000173  013246            JMP     @GENSEL(R3)
1692 013246  013266          GENSEL: GEN0                        ;ALL ZERO WORD
1693 013250  013272                  GEN1                        ;ALL ONE WORD
1694 013252  013300                  GEN52                       ;52 PATTERN
1695 013254  013306                  GEN25                       ;25 PATTERN
1696 013256  013314                  GENR1                       ;ROTATE '1' EACH CALL
1697 013260  013324                  GENR0                       ;ROTATE '0' EACH CALL
1698 013262  013362                  GENRAN                      ;RANDOM NUMBER
1699 013264  013502                  GENINC                      ;INCREMENTING COUNT
1700 013266  005000          GEN0:   CLR     R0                  ;0>R0
1701 013270  000507                  BR      GENEX
1702 013272  005000          GEN1:   CLR     R0                  ;NOT0>R0
1703 013274  005100                  COM     R0
1704 013276  000504                  BR      GENEX
1705 013300  012700  052525    GEN52:  MOV     #52525,R0         ;5252>R0
1706 013304  000501                  BR      GENEX
1707 013306  012700  125252    GEN25:  MOV     #125252,R0        ;125252>R0
1708 013312  000476                  BR      GENEX
1709 013314  000241          GENR1:  CLC
1710 013316  004737  013336            JSR     PC,GENROT         ;SHIFT 1 > R0
1711 013322  000472                  BR      GENEX
1712 013324  000241          GENR0:  CLC
1713 013326  004737  013336            JSR     PC,GENROT         ;
1714 013332  005100                  COM     R0                  ;SHIFT 0 > R0
1715 013334  000465                  BR      GENEX
1716 013336  006037  013360    GENROT: ROR     GENISH            ;ROTATE  1 PATTERN
1717 013342  001003                  BNE     GENER1              ;= 0?
1718 013344  012737  100000 013360    MOV     #100000,GENISH     ;YES, SET MSB
1719 013352  013700  013360    GENER1: MOV     GENISH,R0         ;PUT 1 IN R0
1720 013356  000207                  RTS     PC                  ;AND EXIT
1721 013360  000001          GENISH: 1
1722 013362  012737  000005 002402    GENRAN: MOV     #5,RANSEL  ;SET SELECT VALUE TO 5
1723 013370  004737  013402            JSR     PC,RANGEN         ;GENERATE RANDOM NUMBER IN R0
1724 013374  013700  002406            MOV     RANDN,R0
1725 013400  000443                  BR      GENEX               ;
1726 013402  013702  002406    RANGEN: MOV     RANDN,R2          ;
1727 013406  001002                  BNE     RAN1                ;IS RANDOM = 0
1728 013410  013702  002400            MOV     RANST,R2          ;YES, PUT RANDOM START VALUE IN
1729 013414  032737  000777 002402    RAN1:   BIT     #777,RANSEL ;NO;IS RANSEL SELECT VALUE = 0
1730 013422  001003                  BNE     RAN2                ;NO
1731 013424  012737  000001 002402    MOV     #1,RANSEL          ;YES: SET RANSEL = 1
1732 013432  013703  002402    RAN2:   MOV     RANSEL,R3         ;
1733 013436  013702  002406            MOV     RANDN,R2
1734 013442  033702  002404            BIT     RANMTA,R2         ;GET R2 <0 AND 1>
1735 013446  001405                  BEQ     RANCLC              ;
1736 013450  005102                  COM     R2
1737 013452  033702  002404            BIT     RANMTA,R2
1738 013456  001401                  BEQ     RANCLC
1739 013460  000402                  BR      RANSEC
1740 013462  000241          RANCLC: CLC
1741 013464  000401                  BR      RAN4
1742 013466  000261          RANSEC: SEC
```

KMV11 A/B LOGIC DIAG   MACRO M1200   06-JAN-83 09:39   PAGE 30-2
NUMBER GENERATOR

```
1743 013470  006037  002406          RAN4:    ROR    RANDN       ;ROTATE C TO B15
1744 013474  005303                           DEC    R3          ;IS THIS NUMBER REQUIRED?
1745 013476  001357                            BNE    RAN2+4      ;NO, GET ANOTHER
1746 013500  000207                   RANEX:   RTS    PC          ;YES, EXIT
1747 013502  013700  002330          GENINC:  MOV    GOOD,R0      ;INCREMENTS LOC. 'GOOD'
1748 013506  005200                            INC    R0
1749 013510  010037  002330          GENEX:   MOV    R0,GOOD
1750 013514  004737  013610                    JSR    PC,RSTREG
1751 013520  013737  002330  012436           MOV    GOOD,DATA
1752 013526  000207                            RTS    PC
1753
```

```
1755                                    .SBTTL  SAVE REGISTERS
1756                            ;
1757                            ;
1758                            ;
1759                            ;   DESCRIPTION:
1760                            ;
1761                            ;
1762                            ;       ROUTINE TO SAVE ALL THE GENERAL PURPOSE
1763                            ;   REGISTERS ON THE STACK, AND LEAVE THE ADDRESS OF THE
1764                            ;   CALLING ROUTINE ON THE STACK. THE ROUTINE WILL RUN AT
1765                            ;   PRIORITY 7 TO AVOID ANY INTERRUPTS
1766                            ;
1767                            ;   CAUTION:REGISTER RO IS NOT SAVED
1768                            ;
1769                            ;
1770                            ;
1771                            ;   CALLING SEQUENCE:
1772                            ;
1773                            ;           JSR     PC,SAVREG
1774                            ;
1775                            ;   INPUT PARAMETERS:
1776                            ;
1777                            ;       NONE
1778                            ;
1779                            ;
1780                            ;   IMPLICIT INPUT PARAMETERS:
1781                            ;
1782                            ;       NONE
1783                            ;
1784                            ;
1785                            ;   OUTPUT PARAMETERS:
1786                            ;
1787                            ;       REGISTERS 0 THRU 5 ARE SAVED ON THE STACK
1788                            ;   AND THE RETURN ADDRESS OF THE CALLING ROUTINE IS
1789                            ;   SET AS THE LAST ENTRY ON THE STACK
1790                            ;
1791                            ;
1792                            ;
1793                            ;   IMPLICIT OUTPUT PARAMETERS:
1794                            ;
1795                            ;       NONE
1796                            ;
1797                            ;
1798                            ;   COMPLETION CODES:
1799                            ;
1800                            ;       NONE
1801                            ;
1802                            ;
1803                            ;   POSSIBLE ERROR CODES:
1804                            ;
1805                            ;
1806                            ;       NONE
1807                            ;
1808                            ;
1809 013530             SAVREG: GETPRI  SAVSTA
1810 013536                     SETPRI  MAXPRI
1811 013544  012637  002310     MOV     (SP)+,SAVPC    ;SAVE PC FOR RETURN FROM THIS ROUTINE
```

KMV11 A/B LOGIC DIAG   MACRO M1200   06-JAN-83 09:39   PAGE 31-1
SAVE REGISTERS

```
1812 013550  012637 002410          MOV     (SP)+,SAVPC1
1813 013554  010546                 MOV     R5,-(SP)
1814 013556  010446                 MOV     R4,-(SP)
1815 013560  010346                 MOV     R3,-(SP)
1816 013562  010246                 MOV     R2,-(SP)
1817 013564  010146                 MOV     R1,-(SP)
1818 013566  010046                 MOV     R0,-(SP)
1819 013570  013746 002410          MOV     SAVPC1,-(SP)
1820 013574  013746 002310          MOV     SAVPC,-(SP)      ;PUT PC READY FOR
1821 013600                         SETPRI  SAVSTA
1822 013606  000207                 RTS     PC               ;RETURN
1823
1824
1825
```

```
1827                                    .SBTTL   RESTORE REGISTERS
1828                             :
1829                             :
1830                             :
1831                             :       DESCRIPTION:
1832                             :
1833                             :          RESTORE TO RESTORE THE GENERAL PURPOSE
1834                             :       REGISTERS. THE STACK IS LEFT IN THE SAME STATE AS IT
1835                             :       WAS WHEN SAVREG WAS CALLED.
1836                             :
1837                             :       CAUTION:  REGISTER R0 IS NOT SAVED
1838                             :
1839                             :
1840                             :
1841                             :
1842                             :       CALLING SEQUENCE:
1843                             :
1844                             :          JSR    PC,RSTREG
1845                             :
1846                             :
1847                             :       INPUT PARAMETERS:
1848                             :
1849                             :          NONE
1850                             :
1851                             :
1852                             :       IMPLICIT INPUT PARAMETERS:
1853                             :
1854                             :          NONE
1855                             :
1856                             :
1857                             :       OUTPUT PARAMETERS:
1858                             :
1859                             :          R1 THRU R5 RESTORED
1860                             :
1861                             :
1862                             :       IMPLICIT OUTPUT PARAMETERS:
1863                             :
1864                             :          NONE
1865                             :
1866                             :
1867                             :       COMPLETION CODES:
1868                             :
1869                             :          NONE
1870                             :
1871                             :
1872                             :       POSSIBLE ERROR CODES:
1873                             :
1874                             :          NONE
1875                             :
1876                             :
1877                             :
1878 013610              RSTREG: GETPRI   SAVSTA
1879 013616                      SETPRI   MAXPRI
1880 013624  012637  002310      MOV      (SP)+,SAVPC
1881 013630  012637  002410      MOV      (SP)+,SAVPC1
1882 013634  012600              MOV      (SP)+,R0
1883 013636  012601              MOV      (SP)+,R1
```

```
1884 013640 012602                       MOV     (SP)+,R2
1885 013642 012603                       MOV     (SP)+,R3
1886 013644 012604                       MOV     (SP)+,R4
1887 013646 012605                       MOV     (SP)+,R5
1888 013650 013746 002410                MOV     SAVPC1,-(SP)
1889 013654 013746 002310                MOV     SAVPC,-(SP)      ;PUT PC READY FOR
1890 013660                              SETPRI  SAVSTA
1891 013666 000207                       RTS     PC
```

```
1893                                    ;CHECK CONTENT OF ONE OF THE 8 REGISTERS
1894
1895                                    ; CALLING SEQUENCE
1896                                    ;       JSR     R5,CKSELN                      ; N = REGISTER NUMBER
1897                                    ;       .WORD A                     A=EXPECTED CONTENT OF REGISTER N
1898
1899                                    ;OUTPUT PARAMETER:
1900                                    ;       BRANCH IN PC+2 IF ERROR DETECTED
1901                                    ;       BRANCH IN PC IF NO ERROR DETECTED
1902
1903
1904
1905
1906
1907 013670   012537   002330          CKSEL0: MOV     (R5)+,GOOD       ;WRITE GOOD
1908 013674   017737   176610   002354          MOV     @KMVCSR,SEL0     ;READ SEL 0
1909 013702   023737   002354   002330          CMP     SEL0,GOOD        ;CMP ?
1910 013710   001001                            BNE     1$
1911 013712   000402                            BR      2$
1912 013714   062705   000002          1$:      ADD     #2,R5
1913 013720   000205                   2$:      RTS     R5
1914
1915
1916
1917
1918
1919
1920
1921 013722   005037   002330          CBSEL0: CLR     GOOD
1922 013726   012537   002330                  MOV     (R5)+,GOOD
1923 013732   117737   176552   012434          MOVB    @KMVCSR,BSEL0
1924 013740   123737   012434   002330          CMPB    BSEL0,GOOD
1925 013746   001001                            BNE     1$
1926 013750   000402                            BR      2$
1927 013752   062705   000002          1$:      ADD     #2,R5
1928 013756   000205                   2$:      RTS     R5
```

KMV11 A/B LOGIC DIAG   MACRO M1200   06-JAN-83 09:39 PAGE 34
RESTORE REGISTERS

```
1930                                    ;ROUTINE TO CHECK ALL REGISTER FROM SEL0 TO SEL16
1931
1932
1933                                    ;CALLING SEQUENCE:
1934                                    ;        JSR R5,CKALL
1935                                    ;        .WOPD A              A = EXPECTED VALUE FOR SEL0
1936                                    ;        .WORD B              B       "          "      SEL2
1937                                    ;        .WORD C              C       "          "      SEL4
1938                                    ;        .WORD D              D       "          "      SEL6
1939                                    ;        .WORD E              E       "          "      SEL10
1940                                    ;        .WORD F              F       "          "      SEL12
1941                                    ;        .WORD G              G       "          "      SEL14
1942                                    ;        .WORD H              H       "          "      SEL16
1943
1944
1945                                    ;OUTPUT PARAMETER:
1946                                    ;        BRANCH IN PC+2 IF ERROR
1947                                    ;        BRANCH IN PC IF NO ERROR
1948
1949
1950
1951 013760  012537  002332    CKALL:  MOV      (R5)+,GOOD0
1952 013764  012537  002336            MOV      (R5)+,GOOD2
1953 013770  012537  002340            MOV      (R5)+,GOOD4
1954 013774  012537  002342            MOV      (R5)+,GOOD6
1955 014000  012537  002344            MOV      (R5)+,GOOD10
1956 014004  012537  002346            MOV      (R5)+,GOOD12
1957 014010  012537  002350            MOV      (R5)+,GOOD14
1958 014014  012537  002352            MOV      (R5)+,GOOD16
1959
1960 014020  017737  176464  002354    MOV      @KMVCSR,SEL0          ;READ SEL0
1961 014026  000240                     NOP
1962 014030  000240                     NOP
1963 014032  000240                     NOP
1964 014034  017737  176452  002360    MOV      @KMVP02,SEL2          ;READ SEL2
1965 014042  000240                     NOP
1966 014044  000240                     NOP
1967 014046  000240                     NOP
1968 014050  017737  176440  002362    MOV      @KMVP04,SEL4          ;READ SEL4
1969 014056  000240                     NOP
1970 014060  000240                     NOP
1971 014062  000240                     NOP
1972 014064  017737  176426  002364    MOV      @KMVP06,SEL6          ;READ SEL6
1973 014072  000240                     NOP
1974 014074  000240                     NOP
1975 014076  000240                     NOP
1976 014100  017737  176414  002366    MOV      @KMVP10,SEL10         ;READ SEL10
1977 014106  000240                     NOP
1978 014110  000240                     NOP
1979 014112  000240                     NOP
1980 014114  017737  176402  002370    MOV      @KMVP12,SEL12         ;READ SEL12
1981 014122  000240                     NOP
1982 014124  000240                     NOP
1983 014126  000240                     NOP
1984 014130  017737  176370  002372    MOV      @KMVP14,SEL14         ;READ SEL14
1985 014136  000240                     NOP
1986 014140  000240                     NOP
```

KMV11 A/B LOGIC DIAG   MACRO M1200   06-JAN-83 09:39   PAGE 34-1
RESTORE REGISTERS

```
1987 014142 000240                          NOP
1988 014144 017737  176356 002374           MOV     @KMVP16,SEL16           ;READ SEL16
1989
1990
1991 014152 023737  002354 002332           CMP     SEL0,GOOD0
1992 014160 001035                           BNE     1$
1993 014162 023737  002360 002336           CMP     SEL2,GOOD2
1994 014170 001031                           BNE     1$
1995 014172 023737  002362 002340           CMP     SEL4,GOOD4
1996 014200 001025                           BNE     1$
1997 014202 023737  002364 002342           CMP     SEL6,GOOD6
1998 014210 001021                           BNE     1$
1999 014212 023737  002366 002344           CMP     SEL10,GOOD10
2000 014220 001015                           BNE     1$
2001 014222 023737  002370 002346           CMP     SEL12,GOOD12
2002 014230 001011                           BNE     1$
2003 014232 023737  002372 002350           CMP     SEL14,GOOD14
2004 014240 001005                           BNE     1$
2005 014242 023737  002374 002352           CMP     SEL16,GOOD16
2006 014250 001001                           BNE     1$
2007
2008 014252 000402                           BR      2$
2009 014254 062705  000002           1$:     ADD     #2,R5
2010 014260 000205                    2$:     RTS     R5
```

```
2012                                    ;ROUTINE TO CHECK SEL2 TO SEL16
2013
2014
2015
2016
2017
2018 014262  012537  002336       CKREG:  MOV     (R5)+,GOOD2
2019 014266  012537  002340               MOV     (R5)+,GOOD4
2020 014272  012537  002342               MOV     (R5)+,GOOD6
2021 014276  012537  002344               MOV     (R5)+,GOOD10
2022 014302  012537  002346               MOV     (R5)+,GOOD12
2023 014306  012537  002350               MOV     (R5)+,GOOD14
2024 014312  012537  002352               MOV     (R5)+,GOOD16
2025
2026
2027 014316  017737  176170  002360       MOV     @KMVP02,SEL2
2028 014324  000240                        NOP
2029 014326  000240                        NOP
2030 014330  000240                        NOP
2031 014332  000240                        NOP
2032 014334  017737  176154  002362       MOV     @KMVP04,SEL4
2033 014342  000240                        NOP
2034 014344  000240                        NOP
2035 014346  000240                        NOP
2036 014350  000240                        NOP
2037 014352  017737  176140  002364       MOV     @KMVP06,SEL6
2038 014360  000240                        NOP
2039 014362  000240                        NOP
2040 014364  000240                        NOP
2041 014366  000240                        NOP
2042 014370  017737  176124  002366       MOV     @KMVP10,SEL10
2043 014376  000240                        NOP
2044 014400  000240                        NOP
2045 014402  000240                        NOP
2046 014404  000240                        NOP
2047 014406  017737  176110  002370       MOV     @KMVP12,SEL12
2048 014414  000240                        NOP
2049 014416  000240                        NOP
2050 014420  000240                        NOP
2051 014422  000240                        NOP
2052 014424  017737  176074  002372       MOV     @KMVP14,SEL14
2053 014432  000240                        NOP
2054 014434  000240                        NOP
2055 014436  000240                        NOP
2056 014440  000240                        NOP
2057 014442  017737  176060  002374       MOV     @KMVP16,SEL16
2058
2059
2060
2061
2062 014450  023737  002360  002336       CMP     SEL2,GOOD2
2063 014456  001031                        BNE     1$
2064 014460  023737  002362  002340       CMP     SEL4,GOOD4
2065 014466  001025                        BNE     1$
2066 014470  023737  002364  002342       CMP     SEL6,GOOD6
2067 014476  001021                        BNE     1$
2068 014500  023737  002366  002344       CMP     SEL10,GOOD10
```

KMV11 A/B LOGIC DIAG   MACRO M1200 06-JAN-83 09:39 PAGE 35-1
RESTORE REGISTERS

```
2069 014506  001015                      BNE    1$
2070 014510  023737  002370  002346      CMP    SEL12,GOOD12
2071 014516  001011                      BNE    1$
2072 014520  023737  002372  002350      CMP    SEL14,GOOD14
2073 014526  001005                      BNE    1$
2074 014530  023737  002374  002352      CMP    SEL16,GOOD16
2075 014536  001001                      BNE    1$
2076 014540  000402                      BR     2$
2077
2078 014542  062705  000002        1$:   ADD    #2,R5
2079 014546  000205                2$:   RTS    R5
```

```
2081                                        ;ROUTINE TO CLEAR KMV11 MODULE
2082
2083
2084                                        ;CALLING SEQUENCE:
2085                                        ;      JSR PC,CLRKMV
2086
2087                                        ;ROUTINE DESCRIPTION: CLEAR ALL CSR'S REGISTERS AND CHECK IF = 0
2088
2089
2090
2091 014550  005077  175734    CLRKMV: CLR     @KMVCSR
2092 014554  012777  054000  175726        MOV     #MAINT0,@KMVCSR           ;SET MAINTENANCE MODE
2093 014562                                WAITA   0
2094
2095 014574  012702  000010                MOV     #10,R2
2096 014600  013701  012510                MOV     KMVCSR,R1                ;LOAD ADDRESS
2097 014604  005021          1$:           CLR     (R1)+                    ;CLEAR
2098 014606  000240                        NOP
2099 014610  000240                        NOP
2100 014612  000240                        NOP
2101 014614  005302                        DEC     R2                       ;ALL DONE
2102 014616  001372                        BNE     1$                       ;NO
2103 014620  004537  013760                JSR     R5,CKALL                 ;CHECK ALL REG = 0
2104 014624  000000                        .WORD   0
2105 014626  000000                        .WORD   0
2106 014630  000000                        .WORD   0
2107 014632  000000                        .WORD   0
2108 014634  000000                        .WORD   0
2109 014636  000000                        .WORD   0
2110 014640  000000                        .WORD   0
2111 014642  000000                        .WORD   0
2112 014644  000404                        BR      2$                       ;OK BRANCH AT END
2113 014646                                ERRHRD  2,EM0002,PRALL           ;CSR'S REGISTERS CAN'T BE CLEARED
2114 014656  000207          2$:           RTS     PC
2115
```

KMV11 A/B LOGIC DIAG   MACRO M1200 06-JAN-83 09:39 PAGE 37
RESTORE REGISTERS

```
2117                              ;ROUTINE TO SET MAINTENANCE MODE 0 ON KMV11
2118
2119
2120
2121
2122                              ;CALLING SEQUENCE:
2123                              ;       JSR PC,MAINMO
2124
2125
2126
2127
2128                              ;MAINTO = MASTER CLEAR=1 +MAINT1=1 +MODE = 1   ;DCT11 = HOLD
2129
2130
2131
2132                              ;TEST DESCRIPTION:SET MAINTENANCE MODE 0 AND CHECK THAT MASTER CLEAR
2133                              ;                   IS RESET BY DCT11 MICRO PROCESSOR
2134                              ;
2135                              ;                   GIVE AN ERROR IF NOT RESET
2136
2137
2138
2139
2140 014660 012777 054000 175622 MAINMO: MOV    #MAINTO,@KMVCSR          ;LOAD MAINTO
2141 014666 012737 177000 002324         MOV    #177000,DELCT1
2142 014674 012737 000001 002326         MOV    #1,DELCT2
2143 014702 004737 012760                JSR    PC,WAIT2                    ;WAIT
2144 014706 004537 013670                JSR    R5,CKSELO       ;CHECK SELO=0 BUT MODE BIT + MAINT1 BIT
2145 014712 014000                       .WORD  14000
2146 014714 000404                       BR     1$
2147 014716                              ERRHRD 3,EM0001,PRSELO
2148 014726 000207              1$:      RTS    PC
```

```
2150                                          ;ROUTINE TO SET MAINT MODE 1 AND CHECK DCT11 CLEAR SEL0 AFTER HAVING DECODED
2151
2152
2153
2154                                          ;CALLING SEQUENCE:
2155                                          ;      JSR PC,MAINM1
2156
2157
2158
2159                                          ;GIVE AN ERROR IF MASTER CLEAR IS NOT CLEAR BY DCT11
2160                                          ;
2161                                          ;MAINT1= MASTER CLEAR=1 + MAINT 1 =0 + MODE = 1 : T11=NOT IN HOLD
2162
2163
2164
2165
2166
2167
2168 014730   005077   175554      MAINM1: CLR     @KMVCSR
2169 014734   000240                       NOP
2170 014736   000240                       NOP
2171 014740   012777   044000   175542     MOV     #MAINT1,@KMVCSR         ;LOAD ADDRESS
2172 014746   012737   177700   002324     MOV     #177700,DELCT1
2173 014754   012737   000001   002326     MOV     #1,DELCT2
2174 014762   004737   012760             JSR     PC,WAIT2
2175 014766   004537   013670             JSR     R5,CKSEL0              ;CHECK SEL0=0 BUT MODE BIT =1
2176 014772   004000                       .WORD   4000
2177 014774   000404                       BR      1$                     ;OK BRANCH
2178 014776                                ERRHRD  4,EM0001,PRSEL0
2179 015006   000207             1$:     RTS     PC
2180
2181
2182
2183
2184
```

```
2186                                    ;ROUTINE TO SET TEST NUMBER ON BSELO
2187
2188
2189
2190
2191                                    ;CALLING SEQUENCE:
2192                                    ;         JSR R5,TSTNUB
2193                                    ;         .WORD   A                      A=TEST MICRO PROGRAM NUMBER
2194
2195
2196
2197
2198
2199
2200 015010  012537  012456            TSTNUB: MOV     (R5)+,NUB
2201 015014  053777  012456  175466            BIS     NUB,@KMVCSR            ;LOAD TEST NUMBER
2202 015022  012737  170000  002324            MOV     #170000,DELCT1
2203 015030  012737  000001  002326            MOV     #1,DELCT2
2204 015036  004737  012760                    JSR     PC,WAIT2              ;WAIT
2205 015042  000205                            RTS     R5
```

```
2207
2208
2209                              ;ROUTINE TO WRITE OR READ ONE OF THE KMV11 REGISTERS
2210
2211
2212
2213                              ;CALLING SEQUENCE:
2214                              ;JSR    R5,WRITE
2215                              ;.WORD  A                              A=ADDRESS TO WRITE
2216                              ;.WORD  B                              B=DATA TO WRITE
2217                              ;
2218                              ;
2219                              ;
2220                              ;JSR    R5,READ
2221                              ;.WORD  A                              A=ADDRESS TO READ
2222                              ;
2223                              ;
2224                              ;
2225                              ;MICRO DIAG NB 47 DESCRIPTION:
2226                              ;WRITE: PUT ADDRESS TO WRITE IN SEL2
2227                              ;       PUT DATA TO WRITE IN SEL4
2228                              ;       SET BIT 0 OF SEL6(WRITE BIT)
2229                              ;       SET TEST NB 44
2230                              ;       KMV11 CLEAR BSELO WHEN DONE
2231                              ;
2232                              ;
2233                              ;READ:  PUT ADDRESS TO READ IN SEL2
2234                              ;       CLEAR BIT 0 IN SEL6
2235                              ;       SET TEST 47
2236                              ;       KMV11 READ ADDRESS IN SEL2 AND CLEAR BSELO WHEN DONE
2237                              ;       THE READ DATA IS LOAD IN LOCATION ''BAD'' AND ''DATA''
2238
2239
2240
2241
2242 015044  012577  175442      WRITE:  MOV    (R5)+,@KMVP02           ;WRITE ADDRESS
2243 015050  012577  175440              MOV    (R5)+,@KMVP04           ;   ''    DATA
2244 015054  012777  000001  175434      MOV    #1,@KMVP06              ;BIT WRITE
2245
2246 015062  004537  015010              JSR    R5,TSTNUB               ;SEND TEST NB 44
2247 015066  000047                      .WORD  47
2248
2249 015070  000205                      RTS    R5                      ;RETURN
2250
2251
2252
2253
2254
2255
2256 015072  012577  175414      READ:   MOV    (R5)+,@KMVP02           ;SET ADDRESS TO READ
2257 015076  005077  175412              CLR    @KMVP04
2258 015102  005077  175410              CLR    @KMVP06
2259
2260 015106  004537  015010              JSR    R5,TSTNUB               ;SEND TEST NB 44
2261 015112  000047                      .WORD  47
2262
2263 015114  000240                      NOP
```

```
KMV11 A/B LOGIC DIAG    MACRO M1200  06-JAN-83 09:39  PAGE 40-1
RESTORE REGISTERS

 2264 015116  000240                    NOP
 2265
 2266
 2267 015120  004737  013152            JSR    PC,TSTERR         ;CHECK BSEL 0
 2268 015124  000410                    BR     1$                ;OK
 2269 015126  000402                    BR     2$
 2270 015130  000401                    BR     2$
 2271 015132  000400                    BR     2$
 2272
 2273 015134                    2$:     ERRHRD 5,EM0024           ;NO KMV ANSWER
 2274 015144  000205                    RTS    R5
 2275
 2276 015146  017737  175342  012432 1$: MOV   @KMVP04,BAD       ;READ DATA IN BAD
 2277 015154  013737  012432  012436    MOV    BAD,DATA          ;READ DATA IN "DATA" LOCATION
 2278
 2279 015162  000205                    RTS    R5
 2280
 2281
 2282
 2283
```

D 5

```
2285                              .MACRO  ROMCLK
2286                              .LIST
2287                                      JSR     R5,.ROMCLK       ;CLOCK INSTRUCTION
2288                              .NLIST
2289                              .ENDM
2290
2291                              .MACRO  ED$CALL XY
2292                                      .LIST
2293                              ;********************** TEST'XY' *******************
2294                                      .NLiST
2295                                      .ENDM
2296
2297
2298
2299                              .MACRO  BADHEAD
2300                                      .RADIX  10
2301                                      ED$CALL \T$TESTNUM+1
2302                                      .RADIX  8
2303                                      .ENDM
2304
2305
2306
```

```
2308                              .SBTTL  GLOBAL ERROR REPORT SECTION
2309
2310                              ;///////////////////////////////////////////////////////////////////
2311                              ;/      THE GLOBAL ERROR REPORT SECTION CONTAINS ERROR MESSAGES
2312                              ;/      THAT ARE USED IN MORE THAN ONE TEST.
2313                              ;///////////////////////////////////////////////////////////////////
2314
2315                              .NLIST BEX
2316
2317
2318
2319 015164     045     116     045  TFM36:  .ASCIZ  /%N%AREGISTER ADDRESS ERROR,ADDR = %06%A,UNIT = %02/
2320
2321 015247     040     102     125  TIM:    .ASCIZ  / BUS TIMEOUT /
2322
2323 015265     115     101     123  EM0001: .ASCIZ  /MASTER CLEAR FAIL TO RESET: DCT11 CAN'T CLEAR MASTER CLEAR /
2324
2325 015361     040     113     115  EM0002: .ASCIZ  / KMV11 REGISTERS CAN'T BE CLEARED /
2326
2327 015424     040     104     101  EM0003: .ASCIZ  / DATA COMPARE ERROR ON KMV11 REGISTER (SEL2 TO SEL16)/
2328
2329 015512     040     104     101  EM0004: .ASCIZ  / DATA COMPARE ERROR ON BSEL0 WHEN ACCESSED BY QBUS/
2330
2331 015575     040     122     105  EM0005: .ASCIZ  / REGISTER SEL2 CAN'T BE ACCESSED CORRECTLY BY MICRO PROGRAM/
2332
2333 015671     105     122     122  EM0006: .ASCIZ  /ERROR WHEN TESTING SEL4,DCT11 CAN'T  ACCESS SEL4 CORRECTLY/
2334
2335 015764     105     122     122  EM0007: .ASCIZ  /ERROR WHEN TESTING SEL6,DCT11 CAN'T ACCESS SEL6 CORRECTLY/
2336
2337 016056     105     122     122  EM0010: .ASCIZ  /ERROR WHEN TESTING SEL10,DCT11 CAN'T ACCESS SEL10 CORRECTLY/
2338
2339 016152     105     122     122  EM0011: .ASCIZ  /ERROR WHEN TESTING SEL12,DCT11 CAN'T ACCESS SEL12 CORRECTLY/
2340
2341 016246     105     122     122  EM0012: .ASCIZ  /ERROR WHEN TESTING SEL14,DCT11 CAN'T ACCESS SEL14 CORRECTLY/
2342
2343 016342     105     122     122  EM0013: .ASCIZ  /ERROR WHEN TESTING SEL16,DCT11 CAN'T ACCESS SEL16 CORRECTLY/
2344
2345 016436     040     104     101  EM0015: .ASCIZ  / DATA COMPARE ERROR IN RAM MEMORY TEST /
2346
2347 016506     040     124     111  EM0016: .ASCIZ  / TIMEOUT DURING DMA TRANSFER /
2348
2349 016544     040     104     101  EM0020: .ASCIZ  / DATA COMPARE ERROR AFTER DMA  TRANSFER INTO KMV11 /
2350
2351 016630     040     104     101  EM0021: .ASCIZ  / DATA COMPARE ERROR AFTER DMA TRANSFER IN BOTH DIRECTION /
2352
2353 016722     111     116     124  EM0022: .ASCIZ  /INTERUPT  OCCUR AT WRONG LEVEL /
2354
2355 016762     116     117     040  EM0023: .ASCIZ  /NO INTERUPT OCCUR /
2356
2357 017005     116     117     040  EM0024: .ASCIZ  /NO ANSWER FROM KMV11 MODULE , MICRO TEST NOT EXECUTED /
2358
2359 017075     124     111     115  EM0025: .ASCIZ  /TIMEOUT DURING KMV11 MICRO TEST /
2360
2361 017136     111     116     124  EM0026: .ASCIZ  /INTERUPT ON DCT11 WITH ILLEGAL VECTOR WHEN ACCESSING BSEL2 /
2362
2363 017232     116     117     040  EM0027: .ASCIZ  /NO KMV11 ANSWER ,DCT11  RECEIVE NO INTERUPT /
2364
```

KMV11 A/B LOGIC DIAG   MACRO M1200 06-JAN-83 09:39 PAGE 42-1
GLOBAL ERROR REPORT SECTION

```
2365 017307    111    114    114  EM0028: .ASCIZ  /ILLEGAL INTERUPT OCCURED /
2366
2367 017341    104    101    124  EM0030: .ASCIZ  /DATA COMPARE ERROR DURING DMA TRANSFER OUT KMV11 /
2368
2369 017423    105    122    122  EM0031: .ASCIZ  /ERROR DURING BYTE ACCES ON KMV11 REGISTERS /
2370
2371 017477    111    116    124  EM0032: .ASCIZ  /INTERUPT ON DCT11 WITH ILLEGAL VECTOR WHEN ACCESSING BSEL0 /
2372
2373 017573    122    101    115  EM0033: .ASCIZ  /RAM MEMORY ERROR WHEN TRANSFERING BUFFER IN DMA /
2374
2375 017654    120    122    117  EM0034: .ASCIZ  /PROM REVISION IS NOT COMPATIBLE WITH DIAGNOSTIC REVISION /
2376
2377 017746    040    103    110  EM0134: .ASCIZ  / CHECK PROM AND DIAGNOSTIC REVISION /
2378
2379 020013    040    040    120  EM0035: .ASCIZ  /  PROM CHECKSUM ERROR /
2380
2381
2382
```

```
2384 020042      045      116      045  MSEL0:  .ASCIZ  /%N%A SEL0 = %06%A  SHOUD BE  = %06%N/
2385
2386 020107      045      116      045  MREG0:  .ASCIZ  /%N%A SEL0 = %06%A    SHOUD BE  = %06/
2387 020152      045      116      045  MREG2:  .ASCIZ  /%N%A SEL2 = %06%A    SHOUD BE  = %06/
2388 020215      045      116      045  MREG4:  .ASCIZ  /%N%A SEL4 = %06%A    SHOUD BE  = %06/
2389 020260      045      116      045  MREG6:  .ASCIZ  /%N%A SEL6 = %06%A    SHOUD BE  = %06/
2390 020323      045      116      045  MREG10: .ASCIZ  /%N%A SEL10= %06%A    SHOUD BE  = %06/
2391 020366      045      116      045  MREG12: .ASCIZ  /%N%A SEL12= %06%A    SHOUD BE  = %06/
2392 020431      045      116      045  MREG14: .ASCIZ  /%N%A SEL14= %06%A    SHOUD BE  = %06/
2393 020474      045      116      045  MREG16: .ASCIZ  /%N%A SEL16= %06%A    SHOUD BE  = %06/
2394
2395
2396 020537      045      116      045  MSEL2:  .ASCIZ  /%N%A SEL2 = %06%A  SHOUD BE  = %06/
2397
2398 020602      045      116      045  MSEL4:  .ASCIZ  /%N%A SEL4 = %06%A  SHOUD BE  = %06/
2399
2400 020645      045      116      045  MSEL10: .ASCIZ  /%N%A SEL10 = %06%A SHOUD BE  =%06/
2401
2402 020707      045      116      045  MRAM1:  .ASCIZ  /%N%A  RAM ADDRESS  =%06%A,EXTENDED ADDRESS  =%06/
2403 020771      045      116      045  MRAM2:  .ASCIZ  /%N%A          BDDAT = %06%A SHOUD CONTENT  = %06/
2404
2405 021051      045      116      045  MINT:   .ASCIZ  /%N%A GOOD= %06%A  BAD=%06/
2406
2407 021103      045      116      045  MBSEL0: .ASCIZ  /%N%A BSEL0 =%06%A  SHOUD BE  =%06/
2408
2409 021145      045      116      045  MINTR:  .ASCIZ  /%N%A DCT11 ILLEGAL INTERUPT WHEN ADDRESS =%06%A IS WRITTEN/
2410
2411 021240      045      116      045  MDMA1:  .ASCIZ  /%N%A    DMA ERROR AT ADDRESS =%06%A EXTADDRESS =%06/
2412 021323      045      116      045  MDMA2:  .ASCIZ  /%N%A      BDDAT=%06%A SHOULD BE =%06/
2413
2414 021371      045      116      045  MBYTE:  .ASCIZ  /%N%A    AT ADDRESS ADDR =%06%A, GOOD =%06%A,BAD =%06/
2415
2416 021453      045      116      045  MDMAR1: .ASCIZ  /%N%A    RAM MEMORY LOCATION  =%06%A  IS MODIFIED DURING /
2417 021542      045      116      045  MDMAR2: .ASCIZ  /%N%A  DMA TRANSFER IN BOTH DIRECTION /
2418 021610      045      116      045  MDMAR3: .ASCIZ  /%N%A      READ DATA  =%06%A  SHOULD BE  =%06/
2419
2420 021666      045      116      045  MCHECK: .ASCIZ  /%N%A    CHECKSUM IS =%06%A SHOULD BE ZERO /
2421
2422 021737      045      116      045  MDMAF:  .ASCIZ  /%N%A    ADDR = %06%A  ,GDDAT = %06%A  ,BDDAT  = %06/
2423
2424
2425
```

```
2427
2428                                    .EVEN
2429
2430
2431
2432                                    ;--------------------------------------------------------
2433                                    ; MACRO'S NEEDED TO REPORT ERRORS
2434                                    ;--------------------------------------------------------
2435
2436 022022                            BGNMSG  PRSELO                          ;REPORT CONTENT OF SELO
2437 022022                            PRINTB  #MSELO,SELO,GOOD
2438 022052  004737  013010            JSR     PC,CHKMAX                       ;CHECK IF MAX ERROR?
2439 022056                            ENDMSG
2440
2441
2442 022060                            BGNMSG  PRRAM                           ;RAM ERROR REPORT
2443 022060                            PRINTB  #MRAM1,ADDR,EXADDR
2444 022110                            PRINTB  #MRAM2,BDDAT,GOOD
2445 022140  004737  013010            JSR     PC,CHKMAX                       ;CHECK IF MAX ERROR?
2446 022144                            ENDMSG
2447
2448
2449
2450 022146                            BGNMSG  PRBYTE                          ;BYTE ACCES REPORT
2451 022146                            PRINTB  #MBYTE,ADDR,GOOD,BAD
2452 022202  004737  013010            JSR     PC,CHKMAX                       ;CHECK IF MAX ERROR?
2453 022206                            ENDMSG
2454
2455
2456 022210                            BGNMSG  PDMARA                          ;DMA IN RAM ERROR REPORT
2457 022210                            PRINTB  #MDMAR1,ADDR
2458 022234                            PRINTB  #MDMAR2
2459 022254                            PRINTB  #MDMAR3,BDDAT,GOOD
2460 022304                            BREAK
2461 022306  004737  013010            JSR     PC,CHKMAX                       ;CHECK IF MAX ERROR?
2462 022312                            ENDMSG
2463
2464
2465 022314                            BGNMSG  PCHECK                          ;CHECKSUM ERROR REPORT
2466 022314                            PRINTB  #MCHECK,BAD
2467 022340  004737  013010            JSR     PC,CHKMAX                       ;CHECK IF MAX ERROR?
2468 022344                            ENDMSG
2469
2470
2471
2472
2473 022346                            BGNMSG  PADFLT                          ;ADDRESS TEST
2474 022346                            PRINTB  #TFM36,ADDR,UNIT
2475 022376  004737  013010            JSR     PC,CHKMAX
2476 022402                            ENDMSG
2477
2478
2479
2480
2481
```

```
2483 022404                        BGNMSG  PRALL                      ;CSR'S CONTENT REPORT
2484 022404                        PRINTB  #MREG0,SEL0,GOOD0
2485 022434                        PRINTB  #MREG2,SEL2,GOOD2
2486 022464                        PRINTB  #MREG4,SEL4,GOOD4
2487 022514                        PRINTB  #MREG6,SEL6,GOOD6
2488 022544                        PRINTB  #MREG10,SEL10,GOOD10
2489 022574                        PRINTB  #MREG12,SEL12,GOOD12
2490 022624                        PRINTB  #MREG14,SEL14,GOOD14
2491 022654                        PRINTB  #MREG16,SEL16,GOOD16
2492 022704                        BREAK
2493 022706  004737  013010        JSR     PC,CHKMAX                  ;CHECK IF MAX ERROR?
2494 022712                        ENDMSG
2495
2496
2497
2498
2499
2500
2501
2502
2503 022714                        BGNMSG  PRREG                      ;CSR'S REPORT BUT SEL0
2504 022714                        PRINTB  #MREG2,SEL2,GOOD2
2505 022744                        PRINTB  #MREG4,SEL4,GOOD4
2506 022774                        PRINTB  #MREG6,SEL6,GOOD6
2507 023024                        PRINTB  #MREG10,SEL10,GOOD10
2508 023054                        PRINTB  #MREG12,SEL12,GOOD12
2509 023104                        PRINTB  #MREG14,SEL14,GOOD14
2510 023134                        PRINTB  #MREG16,SEL16,GOOD16
2511 023164                        BREAK
2512 023166  004737  013010        JSR     PC,CHKMAX                  ;CHECK IF MAX ERROR?
2513 023172                        ENDMSG
2514
2515
2516
2517
2518
2519
2520
2521 023174                        BGNMSG  PBSEL0                     ;BSEL0 REPORT
2522 023174                        PRINTB  #MBSEL0,BSEL0,GOOD
2523 023224  004737  013010        JSR     PC,CHKMAX                  ;CHECK IF MAX ERROR?
2524 023230                        ENDMSG
2525
2526
2527
2528
2529
2530 023232                        BGNMSG  PINTR                      ;INTERUPT REPORT
2531 023232                        PRINTB  #MINTR,ADDR
2532 023256  004737  013010        JSR     PC,CHKMAX                  ;CHECK IF MAX ERROR?
2533 023262                        ENDMSG
2534
2535
2536
```

```
2538
2539 023264                        BGNMSG   PRDMA                            ;DMA ERROR REPORT
2540 023264                        PRINTB   #MDMA1,ADDR,EXADDR
2541 023314                        PRINTB   #MDMA2,BDDAT,GOOD
2542 023344   004737  013010       JSR      PC,CHKMAX                        ;CHECK IF MAX ERROR?
2543 023350                        BREAK
2544 023352                        ENDMSG
2545
2546
2547 023354                        BGNMSG   PDMAF                            ;DMA SHORT REPOR1
2548 023354                        PRINTB   #MDMAF,ADDR,GDDAT,BDDAT
2549 023410                        ENDMSG
2550
2551
2552
2553
```

```
2555                             .SBTTL  REPORT CODING SECTION
2556
2557
2558                             ;++
2559                             ; THE REPORT CODING SECTION CONTAINS THE
2560                             ; "PRINTS" CALLS THAT GENERATE STATISTICAL REPORTS.
2561                             ;--
2562
2563 023412                          BGNRPT
2564
2570
2571 023412                          EXIT    RPT
2572
2579
2580 023416                          ENDRPT
2581
2582
```

```
2584                                    .SBTTL  INITIALIZE SECTION
2585
2586                                    ;///////////////////////////////////////////////////////////////
2587                                    ;/ THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
2588                                    ;/ AT THE BEGINNING OF EACH PASS.
2589                                    ;///////////////////////////////////////////////////////////////
2590
2591 023420                             BGNINIT
2592
2593
2628
2629                                    .EVEN
2630
2631
2632
2633
2634                                    ;INITIALIZE SUBROUTINE STACK
2635 023420  012705  012732                     MOV     #SSTACK,R5
2636                                    ;STORE BASE LEVEL PROGRAM STACK POINTER
2637 023424  010637  002304                     MOV     SP,PSTACK
2638 023430  005737  002316                     TST     FTIME
2639 023434  001011                             BNE     1$
2640 023436  013737  000004  002312             MOV     @#4,SAVE4
2641 023444  013737  000006  002314             MOV     @#6,SAVE6
2642 023452  012737  000001  002316             MOV     #1,FTIME
2643 023460  013737  002312  000004    1$:      MOV     SAVE4,@#4
2644 023466  013737  002314  000006             MOV     SAVE6,@#6
2645
2646 023474                             READEF  #EF.START                      ;START COMMAND?
2647 023502                             BCOMPLETE        SETUP                  ;IF YES BRANCH
2648
2649 023504                             READEF  #EF.CONTINUE                    ;CONTINUE COMMAND?
2650 023512                             BCOMPLETE        END
2651
2652
2653 023514                             READEF  #EF.NEW                         ;NEW PASS?
2654 023522                             BNCOMPLETE       NEXT                   ;IF NOT EXIT SETUP
2655
2656 023524  012737  177777  012472    SETUP:   MOV     #-1,UUT                ;INITIALISE UNIT NUMBER
2657
2658 023532  005237  012472            NEXT:    INC     UUT                    ;POINT NEXT UNIT
2659 023536  023737  012472  002270             CMP     UUT,L$UIT              ;ALL DONE?
2660 023544  001521                             BEQ     ABORT                  ;IF YES END OF PASS
2661
2662 023546  013701  012472                     MOV     UUT,R1
2663 023552                                      PRINTF  #RUNNING,R1
2664                                             .EVEN
2665
2666 023574                             GPHARD  UUT,R1                          ;GET P TABLE
2667 023604                             BNCOMPLETE       NEXT                   ;IF NOT AVAILABLE GET NEXT
2668
2669
2670 023606                             GETPRM:
2671                                                                            ;GET ADDRESS OF KMV11
2672 023606  011137  012510                     MOV     (R1),KMVCSR
2673                                                                            ;GET POINTER TO KMV11 SEL02 REG
2674 023612  011137  012512                     MOV     (R1),KMVP02
```

```
2675 023616  062737  000002  012512         ADD     #2,KMVP02
2676                                                                  ;GET POINTER TO KMV11 PORT REG - SEL 4
2677 023624  011137  012514                 MOV     (R1),KMVP04
2678 023630  062737  000004  012514         ADD     #4,KMVP04
2679                                                                  ;GET POINTER TO KMV11 PORT REG - SEL 6
2680 023636  011137  012516                 MOV     (R1),KMVP06
2681 023642  062737  000006  012516         ADD     #6,KMVP06
2682                                                                  ;GET POINTER TO KMV11 REG 10
2683 023650  011137  012520                 MOV     (R1),KMVP10
2684 023654  062737  000010  012520         ADD     #10,KMVP10
2685                                                                  ;GET POINTER TO KMV11 REG 12
2686 023662  011137  012522                 MOV     (R1),KMVP12
2687 023666  062737  000012  012522         ADD     #12,KMVP12
2688                                                                  ;GET POINTER TO KMV11 REG 14
2689 023674  011137  012524                 MOV     (R1),KMVP14
2690 023700  062737  000014  012524         ADD     #14,KMVP14
2691                                                                  ;GET POINTER TO KMV11 REG 16
2692 023706  012137  012526                 MOV     (R1)+,KMVP16
2693 023712  062737  000016  012526         ADD     #16,KMVP16
2694                                                                  ;GET POINTER TO  VECTOR 0
2695 023720  011137  012474                 MOV     (R1),KMVV00
2696                                                                  ;GET POINTER TO  VECTOR 2
2697 023724  011137  012502                 MOV     (R1),KMVV02
2698 023730  062737  000002  012502         ADD     #2,KMVV02
2699                                                                  ;GET POINTER TO VECTOR 4
2700 023736  011137  012500                 MOV     (R1),KMVV04
2701 023742  062737  000004  012500         ADD     #4,KMVV04
2702                                                                  ;GET POINTER TO VECTOR 6
2703 023750  012137  012504                 MOV     (R1)+,KMVV06
2704 023754  062737  000006  012504         ADD     #6,KMVV06
2705                                                                  ;GET POINTER TO TX PRIORITY LEVEL
2706 023762  012137  012476                 MOV     (R1)+,KMVLVL
2707 023766  062737  000006  012506         ADD     #6,KMTLVL
2708                                                                  ;GET LOOPBACK PARAMETERS:
2709 023774  011137  012530                 MOV     (R1),LOOP
2710
2711 024000  005037  002300                 CLR     ERRCNT           ;CLEAR ERROR COUNT
2712 024004                                  EXIT    INIT
2713
2714
2715
2716 024010                         ABORT:   DOCLN                            ;CLEAN UP AND ABORT PASS
2717 024012                                  EXIT INIT                        ;EXIT
2718
2719                                          .NLIST  BEX
2720 024016      045     116     045 RUNNING:  .ASCIZ /%N%A RUNNING ON UNIT %D2%A  PASS TIME = 2 MINUTES/
2721                                          .LIST   BEX
2722                                          .EVEN
2723
2724
2725
2726 024100                         END:     ENDINIT
2727
2728
2729
2730
```

KMV11 A/B LOGIC DIAG   MACRO M1200 06-JAN-83 09:39 PAGE 49
AUTODROP SECTION

```
2732                                    .SBTTL  AUTODROP SECTION
2733
2734                                    ;++
2735                                    ; THIS CODE IS EXECUTED IMMEDIATELY AFTER THE INITIALIZE CODE IF
2736                                    ; THE "ADR" FLAG WAS SET.  THE UNIT(S) UNDER TEST ARE CHECKED TO
2737                                    ; SEE IF THEY WILL RESPOND.  THOSE THAT DON'T ARE IMMEDIATELY
2738                                    ; DROPPED FROM TESTING.
2739                                    ;--
2740                                    .EVEN
2741 024102                                     BGNAUTO
2742
2749
2750
2751
2752                                    ;CHECK IF EXISTING DEVICE
2753
2754
2755 024102  013701  012510                     MOV     KMVCSR,R1       ;R1 CONTAINS BASE KMV11 ADDRESS
2756 024106  012705  000007                     MOV     #7,R5           ;7 REGISTERS TO BE TESTED
2757 024112  012737  024144  000004             MOV     #2$,4           ;SET OUT TIMEOUT TRAP
2758 024120  012737  000340  000006             MOV     #340,6          ;LEVEL 7
2759 024126  005711                     1$:     TST     (R1)            ;REFERENCE DEVICE REGISTERS
2760 024130  000240                             NOP
2761 024132  062701  000002                     ADD     #2,R1           ;NEXT REGISTER
2762 024136  005305                             DEC     R5              ;DEC REGISTER COUNT
2763 024140  001372                             BNE     1$              ;BR IF NOT LAST REGISTER
2764 024142  000405                             BR      3$
2765
2766 024144  062706  000004             2$:     ADD     #4,SP
2767 024150                                     DODU    LOGDEV
2768
2769 024156  013737  002312  000004     3$:     MOV     SAVE4,4
2770 024164  013737  002314  000006             MOV     SAVE6,6
2771
2772
2773
2774
2775 024172                                     ENDAUTO
2776
2777
2778
2779
```

```
2781                                    .SBTTL  CLEANUP CODING SECTION
2782
2783                            ;////////////////////////////////////////////////////////////////
2784                            ;/ THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED
2785                            ;/ AT THE END OF EACH PASS.
2786                            ;////////////////////////////////////////////////////////////////
2787
2788 024174                                BGNCLN
2789
2790
2810
2811
2812
2813 024174                                BRESET
2814
2815 024176                                ENDCLN
2816
2817
2818
2819
2820
```

```
2822                                    .SBTTL   DROP UNIT SECTION
2823
2824                                    ;////////////////////////////////////////////////////////////////////
2825                                    ;/ THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
2826                                    ;/ TO NO LONGER BE TESTED.
2827                                    ;////////////////////////////////////////////////////////////////////
2828
2829 024200                                      BGNDU
2830
2831
2832
2841
2842
2854
2855                                            .EVEN
2856
2857 024200                                     PRINTF   #DROPD,RO        ;UNIT DROPPED
2858
2859 024222                                     EXIT    DU
2860
2861
2862
2863
2864                                            .NLIST   BEX
2865 024226      045     116     045   DROPD:   .ASCIZ   /%N%A UNIT %D2%A DROPPED/
2866                                            .LIST    BEX
2867                                            .EVEN
2868
2869 024256                                     ENDDU
2870
2871
2872
2873
2874
```

```
2876                              .SBTTL  ADD UNIT SECTION
2877
2878                              ;////////////////////////////////////////////////////////////////
2879                              ;/ THE ADD-UNiT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
2880                              ;/  TO BE (A) TESTED FOR THE FIRST TIME, OR (B) RESUMED IN TESTING.  IF
2881                              ;/  ''EF.AUNIT'' IS SET, THE UNIT WILL BE TESTED AS A NEW UNIT.
2882                              ;////////////////////////////////////////////////////////////////
2883
2884
2885
2894
2895 024260                          BGNAU
2896 024260                          ENDAU
2897
2898
2899
2900
2901
2902
```

```
2904
2905                                .SBTTL   HARDWARE TESTS
2906
2907
2908
2909                                ;START OF CODE BLOCK WHICH IS USED AS DATA
2910 024262                         ROMMAP::++
2911                                ; TEST TO ...
2912                                ;--
2913
2920
2926
2927                                ;       BGNTST
2928
2934
2935                                ;       EXIT    TST
2936
2948
2949                                        .EVEN
2950                                ;       ENDTST
2951
2957
```

```
2959 024262                         BADHEAD
                           ;******************** TEST1 ********************
2960                          ;*VERIFY THAT REFERENCING QBUS DEVICE REGISTERS
2961                          ;*DOES NOT CAUSE A TIME OUT TRAP
2962 024262                         BADHEAD
                           ;******************** TEST1 ********************
2963
2964 024262                 BGNTST
2965 024262  013701  012510         MOV    KMVCSR,R1      ;R1 CONTAINS KMV11 ADDRESSES
2966 024266  012705  000007         MOV    #7,R5          ;7 REGISTERS TO BE TESTED
2967 024272  012737  024330  000004 MOV    #2$,4          ;SET OUT TIMEOUT TRAP
2968 024300  012737  000340  000006 MOV    #340,6         ;LEVEL 7
2969 024306  005711            1$:  TST    (R1)           ;REFERENCE DEVICE REGISTERS
2970 024310  000240              NOP
2971 024312                    ESCAPE TST
2972 024316  062701  000002      ADD    #2,R1          ;NEXT REGISTER
2973 024322  005305              DEC    R5             ;DEC REGISTER COUNT
2974 024324  001370              BNE    1$             ;BR IF NOT LAST REGISTER
2975 024326  000413              BR     3$
2976
2977 024330  062706  000004  2$:  ADD    #4,SP
2978 024334  010137  002420      MOV    R1,ADDR        ;REPORT ADDRESS LOCATION
2979 024340  013737  012472  002272 MOV    UUT,UNIT       ;REPORT UNIT NUMBER
2980 024346                    ERRHRD 0,TIM,PADFLT    ;BUS TIMEOUT,ADDRESS PROBLEM ON THIS UNIT
2981
2982 024356  013737  002312  000004 3$:  MOV   SAVE4,4
2983 024364  013737  002314  000006    MOV   SAVE6,6
2984 024372                    ESCAPE TST
2985
2986 024376                 ENDTST
2987                         .EVEN
2988
```

KMV11 A/B LOGIC DIAG   MACRO M1200 06-JAN-83 09:39 PAGE 55
HARDWARE TESTS

```
2990 024400                          BADHEAD
                                     ;********************* TEST2 *******************
2991                                 ;CLEAR ALL KMV11 REGISTERS AND CHECK
2992 024400                          BADHEAD
                                     ;********************* TEST2 *******************
2993
2994
2995
2996
2997
2998 024400                          BGNTST
2999 024400                          BGNSUB
3000
3001 024402                          RESTST:
3002 024402  005077  166102                  CLR     @KMVCSR
3003 024406  012777  054000  166074          MOV     #MAINTO,@KMVCSR         ;SET MASTER CLEAR TO EXIT
3004                                                                         ;SELF TEST IF RUNNING
3005 024414                                  WAITA   0
3006
3007
3008
3009 024426  012702  000010                  MOV     #10,R2                  ;LOAD NUMBER OF REGISTER
3010 024432  013701  012510                  MOV     KMVCSR,R1
3011 024436  005021                  3$:      CLR     (R1)+                   ;CLR KMV11 REGISTERS
3012 024440  000240                           NOP
3013 024442  000240                           NOP
3014 024444  005302                           DEC     R2
3015 024446  001373                           BNE     3$                      ;ALL DONE?
3016
3017 024450  004537  013760                  JSR     R5,CKALL                ;CHECK ALL REGISTERS = 0
3018 024454  000000                          .WORD   0
3019 024456  000000                          .WORD   0
3020 024460  000000                          .WORD   0
3021 024462  000000                          .WORD   0
3022 024464  000000                          .WORD   0
3023 024466  000000                          .WORD   0
3024 024470  000000                          .WORD   0
3025 024472  000000                          .WORD   0
3026 024474  000406                          BR      2$                      ;OK BRANCH
3027 024476                                   ERRHRD  1,EM0002,PRALL          ;REGISTERS FAIL TO RESET
3028 024506                                   ESCAPE  SUB
3029
3030 024512  000240                  2$:      NOP
3031 024514                          ENDSUB
3032
3033
3034 024516                          BGNSUB
3035 024520  004737  014550                  JSR     PC,CLRKMV               ;CLEAR REGISTERS
3036
3037 024524  012777  054000  165756          MOV     #MAINTO,@KMVCSR         ;SET MASTER CLEAR,MODE BIT AND MAINT
3038
3039 024532                                  WAITA   0
3040
3041 024544  004537  013670                  JSR     R5,CKSELO               ;CHECK MASTER CLR IS RESET BY DCT11
3042 024550  014000                          .WORD   14000
3043
3044 024552  000406                          BR      1$                      ;YES
```

```
3045 024554                        ERRHRD  6,EM0001,PRSELO      ;MASTER CLR FAIL TO RESET
3046 024564                        ESCAPE  SUB
3047
3048 024570              1$:
3049 024570              ENDSUB
3050 024572              ENDTST
3051
3052
3053
3054
```

I 6

KMV11 A/B LOGIC DIAG   MACRO M1200 06-JAN-83 09:39 PAGE 56
HARDWARE TESTS

```
3056 024574                         BADHEAD
                                    ;********************* TEST3 *********************
3057                                ;CHECK QBUS ACCESS ON KMV11 REGISTERS (FROM SEL2 TO SEL16)
3058 024574                         BADHEAD
                                    ;********************* TEST3 *********************
3059
3060
3061
3062 024574                         STARS 1
3063                                ;SET MAINT MODE =0   ;DCT11 DECODE AND GOES IN HOLD
3064                                ;PROCESSOR SEND ROTATING PATTERN TO EACH REGISTERS AND CHECK
3065 024574                         STARS 1
3066
3067
3068
3069
3070
3071 024574                  BGNTST
3072 024574  004737 014550   TSTREG: JSR     PC,CLRKMV            ;CLEAR REGISTERS
3073 024600  004737 014660           JSR     PC,MAINMO            ;SET MAINT MODE 0
3074 024604  012737 000007 002414    MOV     #7,COUNT             ;NUMBER OF REG
3075 024612  012704 024650           MOV     #CHECK,R4
3076 024616  062704 000004           ADD     #4,R4                ;POINT GOOD VALUE OF SEL2
3077 024622  013701 012512           MOV     KMVP02,R1            ;LOAD SEL2 ADDRESS
3078
3079 024626  005003        TSELA:  CLR     R3                   ;SELECT FIRST PATTERN
3080
3081 024630                         BREAK
3082
3083 024632  004737 013230  TSELB:  JSR     PC,GENER             ;GENER PATTERN
3084
3085
3086 024636  013711 012436  1$:     MOV     DATA,(R1)              ;LOAD PATTERN IN REG
3087 024642  013714 012436           MOV     DATA,(R4)              ;LOAD GOOD VALUE
3088 024646  000240                  NOP
3089
3090
3091 024650  004537 014262  CHECK:  JSR     R5,CKREG             ;CHECK ALL REGISTER BUT SEL0
3092 024654  000000                  .WORD   0
3093 024656  000000                  .WORD   0
3094 024660  000000                  .WORD   0
3095 024662  000000                  .WORD   0
3096 024664  000000                  .WORD   0
3097 024666  000000                  .WORD   0
3098 024670  000000                  .WORD   0
3099 024672  000406                  BR      1$                   ;IF GOOD BR
3100 024674                          ERRHRD  7,EM0003,PRREG
3101 024704                          ESCAPE  TST
3102
3103 024710  005203        1$:     INC     R3                   ;NEW PATTERN
3104 024712  022703 000007           CMP     #7,R3                ;ALL DONE
3105 024716  001345                  BNE     TSELB                ;NO BR
3106
3107 024720  005021                  CLR     (R1)+                ;SELECT NEW REG
3108 024722  005024                  CLR     (R4)+                ;POINT NEW GOOD VALUE
3109 024724  005337 002414           DEC     COUNT                ;ALL REG TESTED
3110 024730  001336                  BNE     TSELA                ;NO BR
```

KMV11 A/B LOGIC DIAG   MACRO M1200   06-JAN-83 09:39   PAGE 56-1
HARDWARE TESTS

   3111 024732   ENDTST

KMV11 A/B LOGIC DIAG   MACRO M1200   06-JAN-83 09:39 PAGE 57
HARDWARE TESTS

```
3113 024734                              BADHEAD
                                         ;********************* TEST4 ******************
3114                                     ;CHECK QBUS ACCES ON SELO REGISTER
3115 024734                              BADHEAD
                                         ;********************* TEST4 ******************
3116
3117
3118
3119 024734                              STARS 1
3120                                     ;SET MAINT MODE 0      ;DCT11=HOLD
3121                                     ;SEND ROTATING PATTERN IN SELO (EXCEPT BIT 11,12,14) AND CHECK
3122 024734                              STARS 1
3123
3124
3125
3126
3127 024734                              BGNTST
3128 024734  004737 014550                       JSR     PC,CLRKMV           ;CLEAR REG
3129 024740  004737 014660                       JSR     PC,MAINMO           ;LOAD MAINT MODE 0
3130 024744  005003                              CLR     R3                  ;FIRST PATTERN
3131 024746  012704 025014                       MOV     #CHECK1,R4          ;POINT SEL 0
3132 024752  062704 000004                       ADD     #4,R4
3133 024756  012737 000015  002414  TCSRNB: MOV    #15,COUNT            ;SELECT NB O PATTERN
3134
3135 024764                              BREAK
3136
3137 024766  004737 013230       TCSR:   JSR     PC,GENER            ;GERERATE A  PATTERN
3138 024772  042737 054000  012436        BIC     #54000,DATA         ;MASK   MCLR,MODE,MAINT1
3139 025000  013714 012436               MOV     DATA,(R4)
3140 025004  013777 012436  165476        MOV     DATA,@KMVCSR        ;WRITE PATTERN
3141 025012  000240                       NOP
3142 025014  004537 013760       CHECK1: JSR     R5,CKALL            ;CHECK
3143 025020  000000                       .WORD   0
3144 025022  000000                       .WORD   0
3145 025024  000000                       .WORD   0
3146 025026  000000                       .WORD   0
3147 025030  000000                       .WORD   0
3148 025032  000000                       .WORD   0
3149 025034  000000                       .WORD   0
3150 025036  000000                       .WORD   0
3151 025040  000411                       BR      1$
3152 025042                               ERRHRD  8,EM0004,PRALL
3153 025052                               ESCAPE  TST
3154
3155 025056  005337 002414               DEC     COUNT               ;DONE ENOUGH
3156 025062  001341                       BNE     TCSR
3157
3158
3159 025064  005203               1$:     INC     R3                  ;NEW PATTERN
3160 025066  022703 000007                CMP     #7,R3               ;ALL DONE
3161 025072  001331                       BNE     TCSRNB              ;NO BR
3162 025074                               ENDTST
```

KMV11 A/B LOGIC DIAG   MACRO M1200   06-JAN-83 09:39   PAGE 58
HARDWARE TESTS

```
3164 025076                          BADHEAD
                                     ;********************* TEST5 ********************
3165                                 ;CHECK QBUS BYTE ACCES ON ALL KMV11 REGISTERS
3166 025076                          BADHEAD
                                     ;********************* TEST5 ********************
3167
3168
3169
3170
3171
3172
3173
3174 025076                          STARS 1
3175                                 ;SET MAINT MODE 0   ;DCT11=HOLD
3176                                 ;WRITE PATTERN IN EACH BYTE ON KMV11 REGISTERS AND CHECK
3177                                 ;QBUS SEND VARIOUS PATTERN IN ALL BYTE ADDRESS
3178 025076                          STARS 1
3179
3180
3181
3182
3183
3184 025076                          BGNTST
3185 025076  004737 014550                  JSR    PC,CLRKMV
3186 025102  004737 014660           TBYTE:  JSR    PC,MAINMO         ;SET MAINT MODE0
3187
3188 025106  013701 012510                   MOV    KMVCSR,R1         ;LOAD KMV CSR ADDRESS
3189 025112  012704 000015                   MOV    #15,R4            ;LOAD NUMBER OF REGISTERS
3190
3191 025116  012737 000377 002330   1$:      MOV    #377,GOOD         ;SELECT A PATTERN
3192 025124  142737 000130 002330            BICB   #130,GOOD
3193 025132  153711 002330                   BISB   GOCD,(R1)         ;WRITE 1ST BYTE
3194 025136  005037 012432                   CLR    BAD
3195
3196 025142                                   WAITA  177700
3197
3198 025154                                   BREAK
3199
3200 025156  111137 012432                   MOVB   (R1),BAD          ;READ REG
3201 025162  142737 000130 012432            BICB   #130,BAD          ;MASK UNUSED BITS
3202
3203 025170  123737 002330 012432            CMPB   GOOD,BAD          ;COMPARE
3204 025176  001410                          BEQ    3$                ;IF = BRANCH
3205
3206
3207
3208 025200  010137 002420                   MOV    R1,ADDR           ;PREPARE ERROR REPORT
3209 025204                                   ERRHRD 9,EM0031,PRBYTE   ;DATA CMP ERROR WHEN ACCESSING A BYTE
3210 025214                                   ESCAPE TST
3211
3212
3213
3214 025220  005201               3$:         INC    R1                ;SELECT NEW REGISTER
3215 025222  005304                           DEC    R4                ;DONE ALL?
3216 025224  001334                           BNE    1$
3217
3218
```

KMV11 A/B LOGIC DIAG   MACRO M1200   06-JAN-83 09:39   PAGE 58-1
HARDWARE TESTS

```
3219 025226                          ENDTST
3220
```

KMV11 A/B LOGIC DIAG   MACRO M1200 06-JAN-83 09:39 PAGE 59
HARDWARE TESTS

```
3222 025230                          BADHEAD
                                     ;******************** TEST6 ********************
3223                                 ;DATA TRANSFER ON REGISTER SEL 2
3224 025230                          BADHEAD
                                     ;******************** TEST6 ********************
3225
3226
3227 025230                          STARS 1
3228                                 ;SET MAINT1   ;DCT11 DECODE ,CLEAR SEL0 AND WAIT FOR TEST NUMBER
3229                                 ;THE HOST WRITE A PATTERN IN SEL2
3230                                 ;THE HOST WRITE A TEST NUMBER IN BSEL0
3231                                 ;
3232                                 ;IF DCT11 READ CORRECT VALUE ,IT CLEAR BSEL0
3233                                 ;IF ERROR        SET 100 IN BSEL0 IF DATA CMP ERROR
3234                                 ;                BSEL0 =TST NUMBER IF NO KMV11 ANSWER
3235                                 ;
3236                                 ;
3237                                 ;BSEL0=1 = MICRO DIAG TEST 1  ;DCT11 MUST READ 052525 IN SEL2
3238                                 ;BSEL0=2 = MICRO DIAG TEST 2  ;DCT11 MUST READ 125252 IN SEL2
3239 025230                          STARS 1
3240
3241
3242
3243
3244
3245 025230                          BGNTST
3246 025230  004737 014550              JSR    PC,CLRKMV              ;CLEAR REG
3247 025234  004737 014730              JSR    PC,MAINM1              ;SET MAINT MODE 1
3248
3249 025240                          BGNSUB
3250 025242  012777 052525 165242       MOV    #DATA1,@KMVP02         ;SEND 052525
3251 025250  004537 015010              JSR    R5,TSTNUB             ;SEND    TEST NUMB 1
3252 025254  000001                      .WORD  1
3253 025256  004537 013722              JSR    R5,CBSEL0             ;CHECK BSEL0 = 0
3254 025262  000000                      .WORD  0
3255 025264  000425                      BR     1$                   ;TEST OK BR AT END
3256 025266  004537 013722              JSR    R5,CBSEL0             ;CHECK BSEL0=100
3257 025272  000100                      .WORD  100
3258 025274  000401                      BR     2$
3259 025276  000410                      BR     3$
3260
3261
3262
3263 025300                          2$:    ERRHRD  10,EM0005          ;DATA CMP ERROR
3264 025310  004737 013010              JSR    PC,CHKMAX             ;CHECK IF TOO MANY ERROR
3265 025314                              ESCAPE SUB
3266
3267 025320                          3$:    ERRHRD  11,EM0024          ;NO KMV11 ANSWER
3268 025330  004737 013010              JSR    PC,CHKMAX             ;CHECK IF TOO MANY ERROR
3269 025334                              ESCAPE SUB
3270 025340  000240                  1$:    NOP
3271 025342                          ENDSUB
3272
3273
3274 025344                          BGNSUB
3275 025346  004737 014730              JSR    PC,MAINM1             ;SET MAINT MODE 1
3276 025352  012777 125252 165132       MOV    #DATA2,@KMVP02         ;SEND    125252
```

```
3277 025360  004537  015010              JSR     R5,TSTNUB         ;SEND TEST NUB 2
3278 025364  000002                      .WORD   2
3279 025366  004537  013722              JSR     R5,CBSEL0         ;CHECK DCT11 HAS ANSWERED
3280 025372  000000                      .WORD   0                 ;BY CLEARING SEL0
3281 025374  000425                      BR      1$                ;OK BR
3282 025376  004537  013722              JSR     R5,CBSEL0         ;CHECK IF =100
3283 025402  000100                      .WORD   100
3284 025404  000401                      BR      2$
3285 025406  000410                      BR      3$
3286
3287
3288
3289 025410                      2$:     ERRHRD  12,EM0005         ;DATA CMP ERROR ON SEL2
3290 025420  004737  013010              JSR     PC,CHKMAX         ;CHECK IF TOO MANY ERROR
3291 025424                              ESCAPE  SUB
3292
3293 025430                      3$:     ERRHRD  13,EM0024         ;NO KMV11 ANSWER
3294 025440  004737  013010              JSR     PC,CHKMAX         ;CHECK IF TOO MANY ERROR
3295 025444                              ESCAPE  SUB
3296
3297 025450  000240          1$:         NOP
3298 025452                  ENDSUB
3299 025454                  ENDTST
```

```
3301 025456                        BADHEAD
                                   ;********************* TEST7 *********************
3302                               ; CHECK DATA TRANSFER ON REGISTER SEL4
3303 025456                        BADHEAD
                                   ;********************* TEST7 *********************
3304
3305
3306
3307 025456                        STARS 1
3308                               ;SET MAINT1  ;DCT11 DECODE AND CLEAR SEL0
3309                               ;
3310                               ;THE HOST SEND ROTATING PATTERN IN SEL4,AND SET TEST NUMBER 3 IN BSEL0
3311                               ;
3312                               ;DCT11 READ SEL4 , WRITE CONTENT OF SEL4 INTO SEL2 , CLEAR SEL0 WHEN DONE, AND
3313                               ;WAIT FOR NEW PATTERN
3314                               ;
3315                               ;
3316                               ;
3317                               ;      AFTER TEST        BSEL0=100 IF ERROR DURING TEST
3318                               ;                        BSEL0=0   IF TEST HAS BEEN EXECUTED (IN THAT CASE
3319                               ;                                  CHECK IF CONTENT OF SEL2 IS CORRECT)
3320                               ;
3321                               ;BSEL 0 = 3 ;MICRO DIAG NB 3 ;DCT11 TAKE CONTENT OF SEL4 AND PUT IT IN SEL 2.
3322 025456                        STARS 1
3323
3324
3325 025456                        BGNTST
3326 025456                        BGNSUB
3327 025460  004737 014550           JSR     PC,CLRKMV              ;CLEAR REG
3328 025464  004737 014730           JSR     PC,MAINM1             ;SET MAINT MODE 1
3329 025470  005003                  CLR     R3                    ;SELECT FIRST PATTERN
3330
3331 025472  012737 000005 002414 TGENE1: MOV  #5,COUNT            ;NB OF PATTERN
3332 025500                          BREAK
3333
3334 025502  004737 013230       GENE1:  JSR     PC,GENER           ;GENER 1ST PATTERN
3335 025506  013777 012436 165000         MOV     DATA,@KMVP04       ;LOAD SEL4
3336 025514  004537 015010               JSR     R5,TSTNUB          ;SET TEST NUMB  3
3337 025520  000003                       .WORD   3
3338 025522  004537 013722               JSR     R5,CBSEL0          ;LOOK IF ANSWER
3339 025526  000000                       .WORD   0
3340 025530  000404                       BR      1$                 ;OK BR
3341 025532                               ERRHRD  14,EM0024,PBSEL0   ;BSEL0 NOT CLEARED ,NO  ANSWER
3342 025542  000240           1$:         NOP
3343 025544                        ENDSUB
3344
3345
3346 025546                        BGNSUB
3347 025550  012704 025570           MOV     #TSEL4,R4             ;POINT GOOD VALUE
3348 025554  013764 012436 000006    MOV     DATA,6(R4)
3349 025562  013764 012436 000004    MOV     DATA,4(R4)            ;WRITE GOOD VALUE FOR SEL2 AND SEL4
3350
3351 025570  004537 014262       TSEL4:  JSR     R5,CKREG           ;CHECK SEL2 = SEL4
3352 025574  000000                       .WORD   0
3353 025576  000000                       .WORD   0
3354 025600  000000                       .WORD   0
3355 025602  000000                       .WORD   0
```

```
KMV11 A/B LOGIC DIAG    MACRO M1200   06-JAN-83 09:39   PAGE 60-1
HARDWARE TESTS

    3356 025604  000000                      .WORD    0
    3357 025606  000000                      .WORD    0
    3358 025610  000000                      .WORD    0
    3359 025612  000406                      BR       2$
    3360 025614                              ERRHRD   15,EM0006,PRREG
    3361 025624                              ESCAPE   SUB
    3362
    3363
    3364 025630  005337  002414      2$:     DEC      COUNT              ;DONE ENOUGH?
    3365 025634  001322                      BNE      GENE1
    3366
    3367 025636  005203                      INC      R3                 ;NEW PATTERN
    3368 025640  022703  000007              CMP      #7,R3              ;ALL DONE
    3369 025644  001312                      BNE      TGENE1             ;NO BR
    3370 025646                      ENDSUB
    3371 025650                      ENDTST
```

```
3373 025652                                BADHEAD
                                           ;********************** TEST8 *********************
3374                                        ; CHECK DATA TRANSFER ON REGISTER SEL6
3375 025652                                BADHEAD
                                           ;********************** TEST8 *********************
3376
3377
3378 025652                                STARS 1
3379                                        ;SET MAINT1  ;DCT11 DECODE AND CLEAR SEL0
3380                                        ;
3381                                        ;THE HOST SENDS A ROTATING PATTERN IN SEL6,AND SET TEST NUMBER 4 IN BSEL0
3382                                        ;
3383                                        ;DCT11 READ SEL6 , WRITE CONTENT OF SEL6 IN SEL2 , CLEAR SEL0 WHEN DONE, AND
3384                                        ;WAIT FOR NEW PATTERN
3385                                        ;
3386                                        ;
3387                                        ;
3388                                        ;       AFTER TEST        BSEL0=100 IF ERROR DURING TEST
3389                                        ;                         BSEL0=0   IF TEST HAS BEEN EXECUTED (IN THAT CASE
3390                                        ;                                   CHECK IF CONTENT OF SEL4 IS CORRECT)
3391                                        ;
3392                                        ;BSEL 0 = 4 ;MICRO DIAG NB 4 ;DCT11 TAKE CONTENT OF SEL6 AND PUT IT IN SEL 2.
3393 025652                                STARS 1
3394
3395
3396
3397 025652                                BGNTST
3398 025652                                BGNSUB
3399 025654  004737  014550                    JSR     PC,CLRKMV            ;CLEAR REG
3400 025660  004737  014730                    JSR     PC,MAINM1            ;SET MAINT MODE 1
3401 025664  012703  000001                    MOV     #1,R3                ;SELECT 1ST PATTERN
3402 025670  012737  000005  002414 TGENE2:    MOV     #5,COUNT             ;    "    NUMBER OF PATTERN
3403 025676                                    BREAK
3404
3405
3406 025700  004737  013230        GENE2:      JSR     PC,GENER             ;GENERATE 1ST PATTERN
3407 025704  013777  012436  164604            MOV     DATA,@KMVP06         ;LOAD SEL6
3408 025712  004537  015010                    JSR     R5,TSTNUB            ;SET TEST NUMB 4
3409 025716  000004                            .WORD   4
3410 025720  004537  013722                    JSR     R5,CBSEL0            ;LOOK IF ANSWER
3411 025724  000000                            .WORD   0
3412 025726  000404                            BR      1$                   ;OK BR
3413 025730                                    ERRHRD  16,EM0024,PBSEL0     ;NO KMV11 ANSWER,BSEL0 NOT = 0
3414 025740  000240                1$:         NOP
3415 025742                                ENDSUB
3416
3417
3418 025744                                BGNSUB
3419 025746  012704  025766                    MOV     #TSEL6,R4            ;POINT GOOD VALUE
3420 025752  013764  012436  000010            MOV     DATA,10(R4)
3421 025760  013764  012436  000004            MOV     DATA,4(R4)           ;WRITE GOOD VALUE FOR SEL2 AND SEL6
3422
3423 025766  004537  014262        TSEL6:      JSR     R5,CKREG             ;CHECK SEL2 = SEL6
3424 025772  000000                            .WORD   0
3425 025774  000000                            .WORD   0
3426 025776  000000                            .WORD   0
3427 026000  000000                            .WORD   0
```

KMV11 A/B LOGIC DIAG   MACRO M1200 06-JAN-83 09:39 PAGE 61-1
HARDWARE TESTS

```
3428 026002 000000                      .WORD   0
3429 026004 000000                      .WORD   0
3430 026006 000000                      .WORD   0
3431 026010 000406                      BR      2$
3432 026012                             ERRHRD  17,EM0007,PRREG
3433 026022                             ESCAPE  SUB
3434 026026 005337 002414      2$:      DEC     COUNT
3435 026032 001322                      BNE     GENE2
3436
3437 026034 005203                      INC     R3           ;NEW PATTERN
3438 026036 022703 000006               CMP     #6,R3        ;ALL DONE
3439 026042 001312                      BNE     TGENE2       ;NO BR
3440 026044                     ENDSUB
3441 026046                     ENDTST
```

```
3443 026050                           BADHEAD
                                      ;******************* TEST9 *******************
3444                                  ;TEST TO CHECK DATA TRANSFER ON REGISTER SEL10
3445 026050                           BADHEAD
                                      ;******************* TEST9 *******************
3446
3447
3448 026050                           STARS 1
3449                                  ;SET MAINT1  ;DCT11 DECODE AND CLEAR SEL0
3450                                  ;
3451                                  ;THE HOST SENDS A ROTATING PATTERN IN SEL10,AND SET TEST NUMBER 5 IN BSEL0
3452                                  ;
3453                                  ;DCT11 READ SEL10 , WRITE CONTENT OF SEL10 IN SEL2 , CLEAR SEL0 WHEN DONE, AND
3454                                  ;WAIT FOR NEW PATTERN
3455                                  ;
3456                                  ;
3457                                  ;
3458                                  ;       AFTER TEST        BSEL0=100 IF ERROR DURING TEST
3459                                  ;                         BSEL0=0   IF TEST HAS BEEN EXECUTED (IN THAT CASE
3460                                  ;                                   CHECK IF CONTENT OF SEL6 IS CORRECT)
3461                                  ;
3462                                  ;BSEL 0 = 5 ;MICRO DIAG NB 5 ;DCT11 TAKE CONTENT OF SEL10 AND PUT IT IN SEL 2.
3463 026050                           STARS 1
3464
3465
3466
3467 026050                           BGNTST
3468 026050                           BGNSUB
3469 026052  004737  014550                   JSR     PC,CLRKMV           ;CLEAR REG
3470 026056  004737  014730                   JSR     PC,MAINM1           ;SET MAINT MODE 1
3471 026062  012703  000001                   MOV     #1,R3               ;SELECT FIRST PATTERN
3472
3473 026066  012737  000005  002414   TGENE3: MOV     #5,COUNT
3474 026074                                   BREAK
3475
3476
3477 026076  004737  013230           GENE3:  JSR     PC,GENER            ;GENERATE A 1ST PATTERN
3478 026102  042737  040000  012436           BIC     #40000,DATA         ;MASK BIT 14
3479 026110  013777  012436  164402           MOV     DATA,@KMVP10        ;LOAD SEL10
3480 026116  004537  015010                   JSR     R5,TSTNUB           ;SET TEST NUMB  5
3481 026122  000005                           .WORD   5
3482 026124  004537  013722                   JSR     R5,CBSEL0           ;LOOK IF ANSWER
3483 026130  000000                           .WORD   0
3484 026132  000406                           BR      1$                  ;OK BR
3485 026134                                   ERRHRD  20,EM0024,PBSEL0    ;NO KMV11 ANSWER
3486 026144                                   ESCAPE  SUB
3487 026150  000240           1$:             NOP
3488 026152                           ENDSUB
3489
3490
3491 026154                           BGNSUB
3492 026156  012704  026176                   MOV     #TSEL10,R4          ;POINT GOOD VALUE
3493 026162  013764  012436  000012           MOV     DATA,12(R4)
3494 026170  013764  012436  000004           MOV     DATA,4(R4)          ;WRITE GOOD VALUE FOR SEL2 AND SEL10
3495
3496 026176  004537  014262           TSEL10: JSR     R5,CKREG            ;CHECK SEL2 = SEL10
3497 026202  000000                           .WORD   0
```

```
3498 026204  000000                      .WORD    0
3499 026206  000000                      .WORD    0
3500 026210  000000                      .WORD    0
3501 026212  000000                      .WORD    0
3502 026214  000000                      .WORD    0
3503 026216  000000                      .WORD    0
3504 026220  000406                      BR       2$
3505 026222                              ERRHRD   21,EM0010,PRREG        ;DATA CMP ERROR IN SEL10
3506 026232                              ESCAPE   SUB
3507
3508 026236  005337  002414     2$:      DEC      COUNT
3509 026242  001315                      BNE      GENE3
3510
3511 026244  005203                      INC      R3                     ;NEW PATTERN
3512 026246  022703  000006              CMP      #6,R3                  ;ALL DONE
3513 026252  001305                      BNE      TGENE3                 ;NO BR
3514 026254                              ENDSUB
3515 026256                              ENDTST
```

KMV11 A/B LOGIC DIAG   MACRO M1200 06-JAN-83 09:39 PAGE 63
HARDWARE TESTS

```
3517 026260                          BADHEAD
                                     ;********************* TEST10 *******************
3518                                 ;TEST TO CHECK DATA TRANSFER ON REGISTER SEL12
3519 026260                          BADHEAD
                                     ;********************* TEST10 *******************
3520
3521
3522 026260                          STARS 1
3523                                  ;SET MAINT1  ;DCT11 DECODE AND CLEAR SEL0
3524                                  ;
3525                                  ;THE HOST SENDS A ROTATING PATTERN IN SEL12,AND SET TEST NUMBER 6 IN BSEL0
3526                                  ;
3527                                  ;DCT11 READ SEL12 , WRITE CONTENT OF SEL12 IN SEL 2 , CLEAR SEL0 WHEN DONE, AND
3528                                  ;WAIT FOR NEW PATTERN
3529                                  ;
3530                                  ;
3531                                  ;
3532                                  ;       AFTER TEST        BSEL0=100 IF ERROR DURING TEST
3533                                  ;                         BSEL0=0   IF TEST HAS BEEN EXECUTED (IN THAT CASE
3534                                  ;                                     CHECK IF CONTENT OF SEL10 IS CORRECT)
3535                                  ;
3536                                  ;BSEL 0 = 6 ;MICRO DIAG NB 6 ;DCT11 TAKE CONTENT OF SEL12 AND PUT IT IN SEL 2.
3537 026260                          STARS 1
3538
3539
3540
3541 026260                          BGNTST
3542 026260                          BGNSUB
3543 026262  004737  014550               JSR     PC,CLRKMV               ;CLEAR REG
3544 026266  004737  014730               JSR     PC,MAINM1               ;SET MAINT MODE 1
3545 026272  012703  000001               MOV     #1,R3
3546
3547 026276  012737  000005  002414  TGENE4: MOV   #5,COUNT
3548 026304                               BREAK
3549
3550
3551 026306  004737  013230        GENE4:  JSR     PC,GENER                ;GENERATE A 1ST PATTERN
3552 026312  013777  012436  164202        MOV     DATA,@KMVP12            ;LOAD SEL12
3553 026320  004537  015010               JSR     R5,TSTNUB               ;SET TEST NUMB  6
3554 026324  000006                       .WORD   6
3555 026326  004537  013722               JSR     R5,CBSEL0               ;LOOK IF ANSWER
3556 026332  000000                       .WORD   0
3557 026334  000406                       BR      1$                      ;OK BR
3558 026336                               ERRHRD  22,EM0024,PBSEL0                 ;NO KMV11 ANSWER
3559 026346                               ESCAPE  SUB
3560 026352  000240                1$:    NOP
3561 026354                               ENDSUB
3562
3563
3564 026356                          BGNSUB
3565 026360  012704  026400               MOV     #TSEL12,R4              ;POINT GOOD VALUE
3566 026364  013764  012436  000014        MOV     DATA,14(R4)
3567 026372  013764  012436  000004        MOV     DATA,4(R4)             ;WRITE GOOD VALUE FOR SEL2 AND SEL12
3568
3569 026400  004537  014262        TSEL12: JSR     R5,CKREG                ;CHECK SEL2 = SEL12
3570 026404  000000                       .WORD   0
3571 026406  000000                       .WORD   0
```

```
KMV11 A/B LOGIC DIAG    MACRO M1200  06-JAN-83 09:39  PAGE 63-1
HARDWARE TESTS

3572 026410  000000                      .WORD  0
3573 026412  000000                      .WORD  0
3574 026414  000000                      .WORD  0
3575 026416  000000                      .WORD  0
3576 026420  000000                      .WORD  0
3577 026422  000406                      BR     2$
3578 026424                              ERRHRD 23,EM0011,PRREG        ;DATA CMP ERROR IN SEL12
3579 026434                              ESCAPE SUB
3580
3581
3582 026440  005337  002414     2$:      DEC    COUNT
3583 026444  001320                      BNE    GENE4
3584
3585 026446  005203                       INC    R3                   ;NEW PATTERN
3586 026450  022703  000006               CMP    #6,R3                ;ALL DONE
3587 026454  001310                       BNE    TGENE4               ;NO BR
3588 026456                      ENDSUB
3589 026460                      ENDTST
```

K 7

KMV11 A/B LOGIC DIAG   MACRO M1200 06-JAN-83 09:39 PAGE 64
HARDWARE TESTS

```
3591 026462                          BADHEAD
                                     ;******************** TEST11 ********************
3592                                 ; CHECK DATA TRANSFER ON REGISTER SEL14
3593 026462                          BADHEAD
                                     ;******************** TEST11 ********************
3594
3595
3596 026462                          STARS 1
3597                                 ;SET MAINT1  ;DCT11 DECODE AND CLEAR SEL0
3598                                 ;
3599                                 ;THE HOST SEND ROTATING PATTERN IN SEL14,AND SET TEST NUMBER 6 IN BSEL0
3600                                 ;
3601                                 ;DCT11 READ SEL14 , WRITE CONTENT OF SEL14 IN SEL 2 , CLEAR SEL0 WHEN DONE, AND
3602                                 ;WAIT FOR NEW PATTERN
3603                                 ;
3604                                 ;
3605                                 ;
3606                                 ;      AFTER TEST      BSEL0=100 IF ERROR DURING TEST
3607                                 ;                     BSEL0=0   IF TEST HAS BEEN EXECUTED (IN THAT CASE
3608                                 ;                               CHECK IF CONTENT OF SEL12 IS CORRECT)
3609                                 ;
3610                                 ;BSEL 0 = 6 ;MICRO DIAG NB 6 ;DCT11 TAKE CONTENT OF SEL14 AND PUT IT IN SEL 12.
3611 026462                          STARS 1
3612
3613
3614 026462                          BGNTST
3615 026462                          BGNSUB
3616 026464  004737  014550              JSR     PC,CLRKMV               ;CLEAR REG
3617 026470  004737  014730              JSR     PC,MAINM1               ;SET MAINT MODE 1
3618 026474  012703  000001              MOV     #1,R3                   ;SELECT FIRST PATTERN
3619
3620 026500  012737  000005  002414  TGENE5: MOV  #5,COUNT               ;SELECT NB OF PATTERN
3621 026506                          BREAK
3622
3623
3624 026510  004737  013230      GENE5:  JSR     PC,GENER                ;GENERATE A  1ST PATTERN
3625 026514  013777  012436  164002      MOV     DATA,@KMVP14            ;LOAD SEL14
3626 026522  004537  015010              JSR     R5,TSTNUB               ;SET TEST NUMB  7
3627 026526  000007                      .WORD   7
3628 026530  004537  013722              JSR     R5,CBSEL0               ;LOOK IF ANSWER
3629 026534  000000                      .WORD   0
3630 026536  000406                      BR      1$                      ;OK BR
3631 026540                          ERRHRD  24,EM0024,PBSEL0            ;NO KMV11 ANSWER
3632 026550                          ESCAPE  SUB
3633
3634 026554  000240              1$:     NOP
3635 026556                          ENDSUB
3636
3637
3638 026560                          BGNSUB
3639 026562  012704  026602              MOV     #TSEL14,R4              ;POINT GOOD VALUE
3640 026566  013764  012436  000016      MOV     DATA,16(R4)
3641 026574  013764  012436  000004      MOV     DATA,4(R4)              ;WRITE GOOD VALUE FOR SEL2 AND SEL4
3642
3643 026602  004537  014262      TSEL14: JSR     R5,CKREG                ;CHECK SEL2 = SEL14
3644 026606  000000                      .WORD   0
3645 026610  000000                      .WORD   0
```

```
3646 026612  000000                         .WORD   0
3647 026614  000000                         .WORD   0
3648 026616  000000                         .WORD   0
3649 026620  000000                         .WORD   0
3650 026622  000000                         .WORD   0
3651 026624  000406                         BR      2$
3652 026626                                 ERRHRD  25,EM0012,PRREG          ;DATA CMP ERROR IN SEL14
3653 026636                                 ESCAPE  SUB
3654
3655
3656 026642  005337  002414        2$:      DEC     COUNT                    ;DONE ENOUGH?
3657 026646  001320                         BNE     GENE5
3658
3659 026650  005203                         INC     R3                       ;NEW PATTERN
3660 026652  022703  000006                 CMP     #6,R3                    ;ALL DONE
3661 026656  001310                         BNE     TGENE5                   ;NO BR
3662 026660                        ENDSUB
3663 026662                        ENDTST
```

```
KMV11 A/B LOGIC DIAG    MACRO M1200  06-JAN-83 09:39  PAGE 65
HARDWARE TESTS

3665 026664                              BADHEAD
                                         ;******************* TEST12 *******************
3666                                     ; CHECK DATA TRANSFER ON REGISTER SEL16
3667 026664                              BADHEAD
                                         ;******************* TEST12 *******************
3668
3669
3670 026664                              STARS 1
3671                                     ;SET MAINT1  ;DCT11 DECODE AND CLEAR SEL0
3672                                     ;
3673                                     ;THE HOST SENDS A ROTATING PATTERN IN SEL16,AND SET TEST NUMBER 7 IN BSEL0
3674                                     ;
3675                                     ;DCT11 READ SEL16 , WRITE CONTENT OF SEL16 IN SEL 2 , CLEAR SEL0 WHEN DONE, AND
3676                                     ;WAIT FOR NEW PATTERN
3677                                     ;
3678                                     ;
3679                                     ;
3680                                     ;       AFTER TEST       BSEL0=100 IF ERROR DURING TEST
3681                                     ;                        BSEL0=0   IF TEST HAS BEEN EXECUTED (IN THAT CASE
3682                                     ;                                  CHECK IF CONTENT OF SEL14 IS CORRECT)
3683                                     ;
3684                                     ;BSEL 0 = 10 ;MICRO DIAG NB 10 ;DCT11 TAKE CONTENT OF SEL16 AND PUT IT IN SEL 2
3685
3686
3687 026664                              BGNTST
3688 026664                              BGNSUB
3689 026666  004737  014550                  JSR    PC,CLRKMV          ;CLEAR REG
3690 026672  004737  014730                  JSR    PC,MAINM1          ;SET MAINT MODE 1
3691 026676  012703  000001                  MOV    #1,R3              ;SELECT FIRST PATTERN
3692
3693 026702  012737  000005  002414  TGENE6: MOV    #5,COUNT
3694 026710                              BREAK
3695
3696
3697 026712  004737  013230      GENE6:     JSR    PC,GENER           ;GENERATE 1ST PATTERN
3698 026716  013777  012436  163602         MOV    DATA,@KMVP16       ;LOAD SEL16
3699 026724  004537  015010                 JSR    R5,TSTNUB          ;SET TEST NUMB  10
3700 026730  000010                         .WORD  10
3701 026732  004537  013722                 JSR    R5,CBSEL0          ;LOOK IF ANSWER
3702 026736  000000                         .WORD  0
3703 026740  000406                         BR     1$                 ;OK BR
3704 026742                                 ERRHRD 26,EM0024,PBSEL0           ;NO KMV11 ANSWER
3705 026752                                 ESCAPE SUB
3706
3707 026756  000240          1$:            NOP
3708 026760                              ENDSUB
3709
3710
3711 026762                              BGNSUB
3712 026764  012704  027004                  MOV    #TSEL16,R4         ;POINT GOOD VALUE
3713 026770  013764  012436  000020          MOV    DATA,20(R4)
3714 026776  013764  012436  000004          MOV    DATA,4(R4)         ;WRITE GOOD VALUE FOR SEL2 AND SEL16
3715
3716 027004  004537  014262      TSEL16:    JSR    R5,CKREG           ;CHECK SEL2 = SEL16
3717 027010  000000                         .WORD  0
3718 027012  000000                         .WORD  0
3719 027014  000000                         .WORD  0
```

```
KMV11 A/B LOGIC DIAG    MACRO M1200   06-JAN-83 09:39   PAGE 65-1
HARDWARE TESTS

3720 027016  000000                              .WORD   0
3721 027020  000000                              .WORD   0
3722 027022  000000                              .WORD   0
3723 027024  000000                              .WORD   0
3724 027026  000406                              BR      2$
3725 027030                                       ERRHRD  27,EM0013,PRREG
3726 027040                                       ESCAPE  SUB
3727
3728
3729 027044  005337  002414      2$:             DEC     COUNT          ;DONE ENOUGH?
3730 027050  001320                              BNE     GENE6
3731
3732 027052  005203                              INC     R3             ;NEW PATTERN
3733 027054  022703  000006                      CMP     #6,R3          ;ALL DONE
3734 027060  001310                              BNE     TGENE6         ;NO BR
3735 027062                       ENDSUB
3736 027064                       ENDTST
3737
3738
3739
```

KMV11 A/B LOGIC DIAG   MACRO M1200   06-JAN-83 09:39 PAGE 66
HARDWARE TESTS

```
3741 027066                         BADHEAD
                                    ;********************* TEST13 *******************
3742                                ; CHECK DATA TRANSFER ON ALL REGISTERS
3743 027066                         BADHEAD
                                    ;********************* TEST13 *******************
3744
3745
3746
3747
3748
3749 027066                         STARS 1
3750                                ;SET MAINT1  DCT11 CLEAR BSEL0
3751                                ;SEND DIFFERENT PATTERN IN SEL2,  SEND TEST 11
3752                                ;DCT11 READ SEL2 AND WRITTE A CALCULATED VALUE IN SEL4 TO SEL16
3753                                ;                (SEL4)=SEL2+SEL2
3754                                ;                (SEL6)=SEL4+SEL2
3755                                ;                (SEL10)=SEL6+SEL2
3756                                ;                (SEL12)=SEL10+SEL2
3757                                ;                (SEL14)=SEL12+SEL2
3758                                ;                (SEL16)=SEL14+SEL2
3759                                ;DCT11 CLEAR BSEL0 WHEN DONE
3760                                ;
3761                                ;=MICRO DIAG NUMBER 11
3762                                ;
3763                                ;
3764
3765
3766 027066                         STARS 1
3767
3768
3769
3770
3771 027066                         BGNTST
3772 027066  004737  014550                 JSR     PC,CLRKMV               ;CLEAR REG
3773 027072  004737  014730                 JSR     PC,MAINM1               ;SET MAINT1
3774
3775
3776 027076  012703  000004                 MOV     #4,R3                   ;PREPARE INCREMENTING PATTERN
3777 027102  012737  000007  002416          MOV     #7,NUMBER               ;SELECT NUMBER FOR DIFFERENT PATTERN
3778
3779
3780
3781 027110  004737  013230         RGALL:  JSR     PC,GENER                ;PREPARE ONE RANDOM PETTERN
3782
3783 027114  013777  012436  163370         MOV     DATA,@KMVP02            ;WRITE PATTERN IN SEL2
3784 027122  013737  012436  002336          MOV     DATA,GOOD2
3785 027130  004537  015010                 JSR     R5,TSTNUB               ;SEND TEST NB11
3786 027134  000011                         .WORD   11
3787
3788
3789 027136  012737  177700  002324         MOV     #177700,DELCT1
3790 027144  004737  013000                 JSR     PC,WAIT1
3791
3792
3793 027150                         BREAK
3794
3795 027152  004537  013722                 JSR     R5,CBSEL0               ;LOOK IF TEST DONE
```

```
3796 027156  000000                           .WORD   0
3797 027160  000406                           BR      1$              ;YES BRANCH
3798 027162                                   ERRHRD  28,EM0024,PBSELO ;NO KMV11 ANSWER
3799 027172                                   ESCAPE  TST
3800
3801                          .
3802 027176  017737  163310  002360   1$:     MOV     @KMVP02,SEL2    ;READ SEL2 TO SEL16
3803 027204  017737  163304  002362           MOV     @KMVP04,SEL4
3804 027212  017737  163300  002364           MOV     @KMVP06,SEL6
3805 027220  017737  163274  002366           MOV     @KMVP10,SEL10
3806 027226  017737  163270  002370           MOV     @KMVP12,SEL12
3807 027234  017737  163264  002372           MOV     @KMVP14,SEL14
3808 027242  017737  163260  002374           MOV     @KMVP16,SEL16
3809
3810
3811 027250  013737  002336  002340           MOV     GOOD2,GOOD4
3812 027256  063737  002336  002340           ADD     GOOD2,GOOD4
3813
3814 027264  013737  002340  002342           MOV     GOOD4,GOOD6      ;WHAT IS GOOD6
3815 027272  063737  002336  002342           ADD     GOOD2,GOOD6
3816
3817 027300  013737  002342  002344           MOV     GOOD6,GOOD10     ;WHAT IS GOOD10
3818 027306  063737  002336  002344           ADD     GOOD2,GOOD10
3819
3820 027314  013737  002344  002346           MOV     GOOD10,GOOD12    ;  "   "  GOOD12
3821 027322  063737  002336  002346           ADD     GOOD2,GOOD12
3822
3823 027330  013737  002346  002350           MOV     GOOD12,GOOD14    ;  "   "  GOOD14
3824 027336  063737  002336  002350           ADD     GOOD2,GOOD14
3825
3826 027344  013737  002350  002352           MOV     GOOD14,GOOD16    ;  "   "  GOOD16
3827 027352  063737  002336  002352           ADD     GOOD2,GOOD16
3828
3829
3830 027360  023737  002336  002360           CMP     GOOD2,SEL2
3831 027366  001031                           BNE     2$
3832 027370  023737  002340  002362           CMP     GOOD4,SEL4
3833 027376  001025                           BNE     2$
3834 027400  023737  002342  002364           CMP     GOOD6,SEL6
3835 027406  001021                           BNE     2$
3836 027410  023737  002344  002366           CMP     GOOD10,SEL10
3837 027416  001015                           BNE     2$
3838 027420  023737  002346  002370           CMP     GOOD12,SEL12
3839 027426  001011                           BNE     2$
3840 027430  023737  002350  002372           CMP     GOOD14,SEL14
3841 027436  001005                           BNE     2$
3842 027440  023737  002352  002374           CMP     GOOD16,SEL16
3843 027446  001001                           BNE     2$
3844 027450  000410                           BR      3$
3845
3846
3847 027452                          2$:       BREAK
3848 027454                                    ERRHRD  29,EM0003,PRREG
3849 027464                                    BREAK
3850 027466                                    ESCAPE  TST
3851
3852
```

```
3853 027472  005337  002416      3$:     DEC    NUMBER              ;ALL PATTERN DONE?
3854 027476                              BREAK
3855 027500  001203                      BNE    RGALL
3856 027502                      ENDTST
```

```
 3858 027504                                 BADHEAD
                                             ;******************** TEST14 ********************
 3859                                        ;KMV11 RAM MEMORY TEST:  MEMORY PATTERN TEST
 3860 027504                                 BADHEAD
                                             ;******************** TEST14 ********************
 3861
 3862
 3863
 3864
 3865 027504                                 STARS 1
 3866                                         ;SET MAINT MODE 1    ;DCT11 DECODE AND CLEAR SEL 0
 3867                                         ;THE HOST WRITES IN SEL2 THE PATTERN TO BE WRTTEN IN ALL MEMORY
 3868                                         ;AND SETS TEST NUMBER TO 13
 3869                                         ;
 3870                                         ;DCT11 WRITE ALL THE MEMORY WITH THIS VALUE,CHECK IF OK AND
 3871                                         ;WHEN DONE CLEAR BSEL0 IF TEST OK
 3872                                         ;
 3873                                         ;IF ERROR SEL2 = EXPECTED VALUE OF LOCATION IN ERROR
 3874                                         ;         SEL4 = READ VALUE OF LOCATION IN ERROR
 3875                                         ;         SEL6 = ADDRESS IN ERROR
 3876                                         ;
 3877                                         ;
 3878                                         ;
 3879                                         ;
 3880                                         ;BSEL0 = 13 ,  MICRO DIAGNOSTIC TEST NUMBER 13
 3881                                         ;            DCT11 SEND PATTERN IN RAM MEMORY AND CHECK
 3882                                         ;
 3883                                         ;            PATTERN DESCRIPTION:
 3884                                         ;               .   ALL      ZERO
 3885                                         ;               .   ALL      ONE
 3886                                         ;               .   10101010 PATERN
 3887                                         ;               .   01010101 PATERN
 3888                                         ;               .   ROTATING 1
 3889                                         ;               .   ROTATING 0
 3890 027504                                 STARS 1                                 .
 3891
 3892
 3893
 3894
 3895 027504                                 BGNTST
 3896 027504  004737  014550                 RAMPAT: JSR    PC,CLRKMV                ;CLEAR REG
 3897 027510  004737  014730                         JSR    PC,MAINM1                ;SET MAINT1
 3898
 3899 027514  005003                                 CLR    R3                       ;SELECT 1ST PATTERN
 3900 027516  005037  002414                          CLR    COUNT
 3901 027522  004737  013230                 4$:      JSR    PC,GENER                 ;MAKE PATTERN
 3902 027526  013777  012436  162756                  MOV    DATA,@KMVP02             ;WRITE PATTERN IN SEL2
 3903
 3904
 3905 027534  004537  015010                          JSR    R5,TSTNUB               ;SET TEST NB 13
 3906 027540  000013                                  .WORD  13
 3907
 3908 027542                                          WAITB  0,1
 3909
 3910
 3911 027562  004737  013152                          JSR    PC,TSTERR               ;CHECK BSEL0=WHICH ERROR?
 3912 027566  000441                                  BR     1$                       ;TEST OK
```

F 8

KMV11 A/B LOGIC DIAG   MACRO M1200 06-JAN-83 09:39 PAGE 67-1
HARDWARE TESTS

```
3913 027570  000420                          BR      2$
3914                                                             ;TIME OUT ERROR
3915 027572  000427                          BR      3$          ;NO KMV11 ANSWER
3916
3917
3918
3919
3920
3921 027574  017737  162716  002420          MOV     @KMVP06,ADDR    ;READ   ADDRESS OF RAM
3922 027602  017737  162704  002422          MOV     @KMVP02,GDDAT   ;READ EXPECTED DATA (GDDAT)
3923 027610  017737  162700  002424          MOV     @KMVP04,BDDAT   ;READ BAD VALUE OF DATA (BDDAT)
3924
3925 027616                                  ERRHRD  30,EM0015,PRRAM ;DATA CMP ERROR ON ONE RAM LOCATION
3926 027626                                  ESCAPE  TST
3927
3928
3929
3930
3931
3932 027632  005037  002330        2$:       CLR GOOD
3933 027636                                  ERRHRD  31,EM0025,PBSELO ;TIMEOUT ERROR
3934 027646                                  ESCAPE  TST
3935
3936
3937
3938
3939
3940 027652  005037  002330        3$:       CLR     GOOD
3941 027656                                  ERRHRD  32,EM0024,PBSELO ;NO KMV11 ANSWER
3942 027666                                  ESCAPE  TST
3943
3944
3945
3946
3947
3948 027672  005237  002414        1$:       INC     COUNT
3949 027676  022737  000015  002414          CMP     #15,COUNT       ;SEND 1 WORDS IN THE SAME PATTERN
3950 027704  001306                           BNE     4$              ;15 WORDS DONE BR
3951
3952 027706  005037  002414                  CLR     COUNT
3953 027712  005203                           INC     R3              ;TRY WITH A NEW PATTERN
3954 027714  022703  000006                  CMP     #6,R3           ;ALL DONE ?
3955 027720  001300                           BNE     4$              ;NO BR
3956 027722                          ENDTST
```

```
3958 027724                                    BADHEAD
                                               ;******************** TEST15 ********************
3959                                           ;KMV11 RAM MEMORY TEST:  MEMORY ADDRESS TEST
3960 027724                                    BADHEAD
                                               ;******************** TEST15 ********************
3961
3962
3963
3964
3965 027724                                    STARS 1
3966                                            ;SET MAINT MODE 1    ;DCT11 DECODE AND CLEAR SEL 0
3967
3968                                            ;THE HOST SETS TEST NB 14 IN BSEL10
3969                                            ;DCT11 WRITE ADDRESS VALUE IN EACH ADDRESS LOCATIONS FOR ALL
3970                                            ;THE KMV11 RAM.(EXEMPLE: 1000=1000,1002=1002.....).
3971                                            ;DCT11 CLEAR BSEL0 IF TEST IS OK
3972                                            ;         BSEL0= 100 IF DATA COMPARE ERROR DURING CHECK
3973
3974                                            ;IF ERROR SEL2 = EXPECTED VALUEOF LOCATION IN ERROR
3975                                            ;         SEL4 = READ VALUE OF LOCATION IN ERROR
3976                                            ;         SEL6 = ADDRESS IN ERROR
3977                                            ;
3978                                            ;
3979                                            ;
3980                                            ;
3981                                            ;BSEL0 = 14 ,  MICRO DIAGNOSTIC TEST NUMBER 14
3982                                            ;         DCT11 WRITE ADDRESS VALUE IN ADDRESS LOCATION AND CHECK
3983 027724                                    STARS 1
3984
3985
3986
3987
3988 027724                                    BGNTST
3989 027724  004737  014550                    RAMADD: JSR    PC,CLRKMV              ;CLEAR REG
3990 027730  004737  014730                            JSR    PC,MAINM1              ;SET MAINT1
3991
3992 027734  004537  015010                            JSR    R5,TSTNUB              ;SET TEST NB 14
3993 027740  000014                                    .WORD  14
3994
3995 027742                                            WAITB  0,1
3996
3997
3998
3999 027762  004737  013152                            JSR    PC,TSTERR              ;CHECK BSEL0
4000 027766  000441                                    BR     1$                     ;TEST OK
4001 027770  000420                                    BR     2$                     ;TIMEOUT ERROR
4002 027772  000427                                    BR     3$                     ;NO KMV11 ANSWER
4003
4004
4005
4006 027774  017737  162516  002420                    MOV    @KMVP06,ADDR           ;READ   ADDRESS OF RAM
4007 030002  017737  162504  002330                    MOV    @KMVP02,GOOD           ;READ EXPECTED DATA (GDDAT)
4008 030010  017737  162500  002424                    MOV    @KMVP04,BDDAT          ;READ BAD VALUE OF DATA (BDDAT)
4009
4010 030016                                            ERRHRD 33,EM0015,PRRAM        ;DATA CMP ERROR ON ONE RAM LOCATION
4011 030026                                            ESCAPE TST
4012
```

KMV11 A/B LOGIC DIAG   MACRO M1200   06-JAN-83 09:39   PAGE 68-1
HARDWARE TESTS

```
4013
4014
4015
4016
4017 030032  005037  002330        2$:     CLR     GOOD
4018 030036                                ERRHRD  34,EM0025,PBSELO         ;TIMEOUT ERROR
4019 030046                                ESCAPE  TST
4020
4021
402
4023
4024 030052  005037  002330        3$:     CLR     GOOD
4025 030056                                ERRHRD  35,EM0024,PBSELO         ;NO KMV11 ANSWER
4026 030066                                ESCAPE  TST
4027 030072  000240              1$:     NOP
4028 030074                      ENDTST
```

KMV11 A/B LOGIC DIAG   MACRO M1200   06-JAN-83 09:39 PAGE 69
HARDWARE TESTS

```
4030 030076                          BADHEAD
                                     ;******************** TEST16 *******************
4031                                 ;KMV11 RAM MEMORY TEST:  MEMORY ADDRESS COMPLEMENT TEST
4032 030076                          BADHEAD
                                     ;******************** TEST16 *******************
4033
4034
4035
4036
4037 030076                          STARS 1
4038                                 ;SET MAINT MODE 1    ;DCT11 DECODE AND CLEAR SEL 0
4039                                 ;SET TEST NUMBER 15   ;DCT11 EXECUTE TEST
4040                                 ;DCT11 WRITE COMPLEMENT ADDRESS VALUE IN EACH ADDRESS LOCATION AND CHECK.
4041                                 ;
4042                                 ;DCT11 CLEAR BSEL0 IF TEST OK AND PUT 100 IN BSEL0 IF DATA COMPARE ERROR
4043                                 ;
4044                                 ;IF ERROR SEL2 = EXPECTED VALUE OF LOCATION IN ERROR
4045                                 ;         SEL4 = READ VALUE OF LOCATION IN ERROR
4046                                 ;         SEL6 = ADDRESS IN ERROR
4047                                 ;
4048                                 ;
4049                                 ;
4050                                 ;
4051                                 ;BSEL0 = 15 ,  MICRO DIAGNOSTIC TEST NUMBER 15
4052                                 ;             DCT11 WRITE COMPL. ADDRESS IN ADDRESS IN RAM MEMORY AND CHECK
4053 030076                          STARS 1
4054
4055
4056
4057
4058 030076                          BGNTST
4059 030076  004737  014550          RAMCAD: JSR     PC,CLRKMV                ;CLEAR REG
4060 030102  004737  014730                  JSR     PC,MAINM1                ;SET MAINT1
4061
4062 030106  004537  015010                  JSR     R5,TSTNUB                ;SET TEST NB 15
4063 030112  000015                          .WORD   15
4064
4065 030114                                  WAITB   0,1
4066
4067
4068
4069 030134  004737  013152                  JSR     PC,TSTERR                ;CHECK  BSEL0,WHICH ERROR
4070 030140  000441                          BR      1$                       ;TEST OK
4071 030142  000420                          BR      2$                       ;TIMEOUT ERROR
4072 030144  000427                          BR      3$                       ;NO KMV11 ANSWER
4073
4074
4075 030146  017737  162344  002420          MOV     @KMVP06,ADDR             ;READ    ADDRESS OF RAM
4076 030154  017737  162332  002330          MOV     @KMVP02,GOOD             ;READ EXPECTED DATA (GDDAT)
4077 030162  017737  162326  002424          MOV     @KMVP04,BDDAT            ;READ BAD VALUE OF DATA (BDDAT)
4078
4079 030170                                  ERRHRD  36,EM0015,PRRAM          ;DATA CMP ERROR ON ONE RAM LOCATION
4080 030200                                  ESCAPE  TST
4081
4082
4083
4084
```

```
4085 030204  005037  002330      2$:     CLR    GOOD
4086 030210                              ERRHRD 37,EM0025,PBSELO          ;TIMEOUT ERROR
4087 030220                              ESCAPE TST
4088
4089
4090
4091
4092
4093 030224  005037  002330      3$:     CLR    GOOD
4094 030230                              ERRHRD 38,EM0024,PBSELO          ;NO KMV11 ANSWER
4095 030240                              ESCAPE TST
4096
4097
4098
4099
4100 030244  000240             1$:     NOP
4101 030246                     ENDTST
4102
4103
```

KMV11 A/B LOGIC DIAG   MACRO M1200   06-JAN-83 09:39 PAGE 70
HARDWARE TESTS

```
4105 030250                              BADHEAD
                                         ;********************** TEST17 ****************
4106                                      ;CHECK PROM REVISION TO SEE IF COMPATIBLE WHITH DIAGNOSTIC
4107 030250                              BADHEAD
                                         ;********************** TEST17 ****************
4108
4109
4110
4111
4112 030250                              STARS 1
4113                                      ;READ LOCATION 2 OF THE PROM (ADDRESS 160002) WHICH CONTENTS PROM VERSION
4114                                      ;      NUMBER
4115                                      ;CHECK IF DIAGNOSTIC AND PROM ARE COMPATIBLE AND GIVE AN ERROR IF NOT
4116 030250                              STARS 1
4117
4118
4119
4120
4121
4122 030250                              BGNTST
4123 030250  004737  014550                      JSR     PC,CLRKMV           ;CLEAR ALL REGISTERS
4124 030254  004737  014730                      JSR     PC,MAINM1           ;SET MAINT MODE
4125
4126
4127 030260  004537  015072              REVPRO: JSR     R5,READ             ;READ LOCATION 160002
4128 030264  160002                              .WORD   160002
4129
4130
4131 030266  023737  012464  012432              CMP     GDREV,BAD           ;LOOK IF COMPATIBLE
4132 030274  001406                              BEQ     1$                  ;YES
4133
4134 030276                                      ERRHRD  39,EM0034           ;REPORT THE ERROR
4135 030306                                      ESCAPE TST
4136 030312                              1$:
4137 030312                              ENDTST
```

```
4139
4140 030314                        BADHEAD
                                   ;******************** TEST18 *******************
4141                               ;PROM CHECKSUM TEST
4142 030314                        BADHEAD
                                   ;******************** TEST18 *******************
4143
4144
4145
4146
4147 030314                        STARS 1
4148                               ;DIAGNOSIC READS ALL PROM'S LOCATIONS AND ADDS THEN TOGETHER
4149                               ;RESULT MUST BE ZERO
4150                               ;
4151                               ;
4152                               ;TEST 33 DESCRIPTION:
4153                               ;DCT11 ADD ALL PROMS LOCATIONS ,IF RESULT IS ZERO=CLEAR BSELO
4154                               ;                              IF CHECKSUM ERROR =SET 100 IN BSELO
4155 030314                        STARS 1
4156
4157
4158
4159
4160
4161
4162
4163
4164 030314                        BGNTST
4165 030314  004737 014550                 JSR     PC,CLRKMV            ;CLEAR REGISTERS
4166 030320  004737 014730                 JSR     PC,MAINM1            ;SET MAINTENANCE MODE
4167
4168
4169 030324  004537 015010        PROMCK:  JSR     R5,TSTNUB
4170 030330  000033                         .WORD   33                  ;SET TEST 33
4171
4172 030332                                 WAITB   0,1
4173
4174 030352  004737 013152                 JSR     PC,TSTERR            ;TEST IF ERROR
4175 030356  000427                         BR      1$                  ;TEST OK
4176 030360  000412                         BR      2$                  ;TIMEOUT ERROR
4177 030362  000417                         BR      3$                  ;NO ANSWER FROM KMV11
4178
4179
4180 030364  017737 162122 012432           MOV     @KMVP02,BAD         ;CHECKSUM ERROR
4181 030372                                 ERRHRD  40,EM0035,PCHECK
4182 030402                                 ESCAPE  TST
4183
4184
4185 030406                         2$:     ERRHRD  41,EM0025            ;TIMEOUT DURING TEST
4186 030416                                 ESCAPE  TST
4187
4188
4189 030422                         3$:     ERRHRD  42,EM0024            ;NO KMV ANSWER
4190 030432                                 ESCAPE  TST
4191
4192
4193 030436  000240               1$:       NOP
```

KMV11 A/B LOGIC DIAG   MACRO M1200   06-JAN-83 09:39   PAGE 71-1
HARDWARE TESTS

```
4194
4195 030440                              ENDTST
4196
4197
4198
4199
4200
4201
4202
4203
```

KMV11 A/B LOGIC DIAG   MACRO M1200 06-JAN-83 09:39 PAGE 72
HARDWARE TESTS

```
4205
4206 030442                              BADHEAD
                                         ;******************** TEST19 ********************
4207                                     ;TEST DMA TRANSFER IN KMV11
4208 030442                              BADHEAD
                                         ;******************** TEST19 ********************
4209
4210
4211
4212
4213 030442                              STARS 1
4214                                     ;SET MAINT1   ;DCT11 DECODE AND CLEAR BSEL0
4215                                     ;LOAD FIRST ADDRESS OF TX TABLE IN SEL12 , TABLE LENGTH IN SEL14,
4216                                     ;TX TABLE EXTENDED ADDRESS IN BSEL 10.
4217                                     ;
4218                                     ;SET TEST NUMBER (16 OR 17)IN BSEL0
4219                                     ;DCT11 EXECUTE THE DMA TRANSFER OF THE TABLE IN KMV11 RAM AND CHECK.
4220                                     ;WHEN DONE CLEAR BSEL0 IF TEST OK
4221                                     ;SET 200 IN BSEL0 IF TIMEOUT DURING TEST
4222                                     ;SET 100 IN BSEL0 IF ERROR DURING TRANSFER
4223                                     ;IN THAT CASE              SEL2=EXPECTED VALUE
4224                                     ;                          SEL4=READ VALUE
4225                                     ;                          SEL6=ADDRESS LOCATION OF ERROR
4226                                     ;                          BSEL10=EXTENDED ADDRESS
4227                                     ;TEST DESCRIPTION: PDP GENERATE AN INCREMENTING PATTERN TABLE OF 1K WORDS
4228                                     ;SEND STARTING ADDRESS AND TABLE LENGTH TO KMV11
4229                                     ;KMV11 START DMA TRANSFER AND CHECK
4230                                     ;
4231
4232                                     ;TEST 16 = TABLE CONTENT INCREMENTING PATTERN FROM 0
4233                                     ;TEST 17 = EACH LOCATION CONTENT ADDRESS VALUE OF LOCATION
4234 030442                              STARS 1
4235
4236
4237
4238
4239 030442                              BGNTST
4240 030442  004737 014550              DMAIN:  JSR     PC,CLRKMV              ;CLEAR REG
4241 030446  004737 014730                      JSR     PC,MAINM1             ;SET MAINT 1
4242 030452                              BGNSUB
4243 030454  012701 002426                      MOV     #TTABLE,R1            ;POINT TX TABLE
4244 030460  005002                              CLR     R2                   ;CLR TABLE
4245 030462  010221                      1$:     MOV     R2,(R1)+             ;MAKE AN INCREMENTING PATTERN FROM 0
4246 030464  005202                              INC     R2
4247 030466  022702 002000                      CMP     #2000,R2             ;TABLE LENGTH=1K WORDS
4248 030472  001373                              BNE     1$
4249
4250
4251
4252 030474  012777 002426 162020              MOV     #TTABLE,@KMVP12       ;SET TX TABLE ADDRESS
4253 030502  012777 002000 162014              MOV     #2000,@KMVP14         ;SET TABLE LENGTH
4254 030510  005077 162004                      CLR     @KMVP10              ;CLEAR EXTENDED ADDRESS
4255 030514  004537 015010                      JSR     R5,TSTNUB            ;SEND TEST NB 16
4256 030520  000016                              .WORD   16
4257
4258
4259 030522                                      WAITB   0,1                  ;WAIT FOR TEST EXCUTION
```

```
4260
4261
4262
4263 030542   004737   013152                    JSR     PC,TSTERR              ;CHECK BSELO ,WHICH ERROR
4264 030546   000444                             BR      2$                     ;TEST OK
4265 030550   000423                             BR      3$                     ;TIMEOUT ERROR
4266 030552   000432                             BR      4$                     ;NO KMV11 ANSWER
4267
4268
4269
4270
4271
4272 030554   017737   161732   002330           MOV     @KMVP02,GOOD           ;READ GOOD DATA
4273 030562   017737   161726   002424           MOV     @KMVP04,BDDAT          ;READ BAD DATA
4274 030570   017737   161722   002420           MOV     @KMVP06,ADDR           ;READ ERROR ADDRESS
4275 030576   117737   161716   012426           MOVB    @KMVP10,EXADDR         ;READ EXTENDED ADDRESS
4276 030604                                      ERRHRD  43,EM0020,PRDMA        ;DATA CMP ERROR DURING DMA IN TX
4277 030614                                      ESCAPE  SUB
4278
4279
4280
4281
4282
4283 030620   005037   002330          3$:       CLR     GOOD
4284 030624                                      ERRHRD  44,EM0016,PBSELO       ;TIMEOUT ERROR
4285 030634                                      ESCAPE  SUB
4286
4287
4288 030640   005037   002330          4$:       CLR     GOOD
4289 030644                                      ERRHRD  45,EM0024,PBSELO       ;NO KMV ANSWER
4290 030654                                      ESCAPE  SUB
4291
4292
4293 030660   000240                  2$:       NOP
4294 030662                            ENDSUB
4295
4296
4297
4298 030664                            BGNSUB
4299 030666   004737   014730                    JSR     PC,MAINM1
4300 030672   012704   002426                    MOV     #TTABLE,R4             ;POINT TTABLE
4301 030676   012702   002000                    MOV     #2000,R2               ;TABLE LENGTH
4302 030702   010401                  1$:       MOV     R4,R1
4303 030704   010124                            MOV     R1,(R4)+         ;TABLE LOCATION CONTENT TABLE LOCATION ADDRESS
4304 030706   005302                            DEC     R2
4305 030710   001374                            BNE     1$
4306
4307
4308 030712   012777   002426   161602           MOV     #TTABLE,@KMVP12        ;SEND TABLE ADDRESS
4309 030720   012777   002000   161576           MOV     #2000,@KMVP14          ;  "     "   LENGTH
4310 030726   004537   015010                    JSR     R5,TSTNUB              ;SET TEST NB 17
4311 030732   000017                             .WORD   17
4312
4313
4314 030734                                      WAITB   0,1                    ;WAIT FOR TEST EXECUTION
4315
4316
```

```
4317 030754 004737 013152              JSR    PC,TSTERR            ;CHECK BSEL0
4318 030760 000444                     BR     2$                   ;TEST OK
4319 030762 000423                     BR     3$                   ;TIMEOUT ERROR
4320 030764 000432                     BR     4$                   ;NO KMV ANSWER
4321
4322
4323
4324
4325 030766 017737 161520 002330       MOV    @KMVP02,GOOD         ;READ GOOD DATA
4326 030774 017737 161514 002424       MOV    @KMVP04,BDDAT        ;  "   BAD   "
4327 031002 017737 161510 002420       MOV    @KMVP06,ADDR         ;  "   ERROR ADDRESS
4328 031010 117737 161504 012426       MOVB   @KMVP10,EXADDR       ;  "   EXTENDED ADDRESS
4329
4330 031016                            ERRHRD 46,EM0020,PRDMA      ;DATA CMP ERROR
4331 031026                            ESCAPE SUB
4332
4333
4334
4335 031032 005037 002330        3$:   CLR    GOOD
4336 031036                            ERRHRD 47,EM0016,PBSEL0     ;TIMEOUT ERROR
4337 031046                            ESCAPE SUB
4338
4339
4340
4341
4342 031052 005037 002330        4$:   CLR    GOOD
4343 031056                            ERRHRD 48,EM0024,PBSEL0     ;NO KMV ANSWER
4344 031066                            ESCAPE SUB
4345
4346
4347
4348
4349 031072 000240              2$:   NOP
4350 031074                            ENDSUB
4351 031076                            ENDTST
```

KMV11 A/B LOGIC DIAG   MACRO M1200 06-JAN-83 09:39 PAGE 73
HARDWARE TESTS

```
4353 031100                              BADHEAD
                                         ;********************* TEST20 *******************
4354                                     ;TEST DMA TRANSFER OUT KMV11
4355 031100                              BADHEAD
                                         ;********************* TEST20 *******************
4356
4357
4358
4359
4360 031100                              STARS 1
4361                                     ;SET MAINT1   ;DCT11 DECODE AND CLEAR BSELO
4362                                     ;LOAD FIRST ADDRESS OF RX TABLE IN SEL12 AND RX TABLE LENGTH IN SEL14
4363                                     ;EXTENDED ADDRESS IN BSEL10
4364
4365                                     ;SET TEST NUMBER 20,21 ;DCT11 EXECUTE TEST
4366                                     ;WHEN DONE CLEAR BSELO IF TEST OK
4367                                     ;SET 200 IN BSELO IF TIMEOUT DURING TEST
4368                                     ;
4369                                     ;
4370                                     ;
4371                                     ;TEST 20 DESCRIPTION: DCT11 SEND IN DMA AN INCREMENTING PATTERN (OF 1K WORDS)
4372                                     ;                     IN HOST MEMORY. THIS PATTERN STARS AT ADDRESS FOUND
4373                                     ;                     IN SEL12 (RX TABLE)
4374                                     ;                     WHEN DONE CLEAR BSELO
4375                                     ;                     HOST CHECK IF THE RECEIVE TABLE IS CORRECT
4376
4377
4378                                     ;TEST 21 DESCRIPTION: IDEM BUT TABLE CONTENT ADDRESS VALUE OF EACH LOCATION
4379 031100                              STARS 1
4380
4381
4382
4383
4384 031100                              BGNTST
4385 031100                              BGNSUB
4386 031102  004737  014550              DMAOUT: JSR    PC,CLRKMV         ;CLEAR REG
4387 031106  004737  014730                      JSR    PC,MAINM1         ;SET MAINT 1
4388 031112  005037  002320                      CLR    FLAG
4389 031116  012701  006426                      MOV    #RTABLE,R1        ;POINT RX TABLE
4390 031122  012702  002000                      MOV    #2000,R2                   ;CLR RX TABLE
4391 031126  005021              1$:             CLR    (R1)+
4392 031130  005302                              DEC    R2
4393 031132  001375                              BNE    1$
4394
4395
4396
4397 031134  012777  006426  161360             MOV    #RTABLE,@KMVP12   ;SET RX TABLE ADDRESS
4398 031142  012777  002000  161354             MOV    #2000,@KMVP14     ;SET TABLE LENGTH
4399 031150  105077  161344                      CLRB   @KMVP10           ;CLEAR EXTENDED ADDRESS
4400 031154  004537  015010                      JSR    R5,TSTNUB         ;SEND TEST NB 20
4401 031160  000020                              .WORD  20
4402
4403
4404
4405 031162                                      WAITB  0,1               ;WAIT FOR TEST EXECUTION
4406
4407
```

```
4408 031202  004737  013152              JSR     PC,TSTERR           ;CHECK  BSELO;WHICH ERROR
4409 031206  000423                      BR      2$                  ;TEST OK
4410 031210  000402                      BR      5$                  ;TIMEOUT ERROR
4411 031212  000411                      BR      6$                  ;NO KMV ANSWER
4412 031214  000420                      BR      2$
4413
4414
4415 031216  005037  002330      5$:     CLR     GOOD
4416 031222                              ERRHRD  49,EM0016,PBSELO    ;TIMEOUT ERROR
4417 031232                              ESCAPE  SUB
4418
4419
4420
4421 031236  005037  002330      6$:     CLR     GOOD
4422 031242                              ERRHRD  50,EM0024,PBSELO    ;NO KMV ANSWER
4423 031252                              ESCAPE  SUB
4424
4425
4426
4427
4428
4429 031256  012701  006426      2$:     MOV     #RTABLE,R1          ;CHECK RX TABLE
4430 031262  005037  002422              CLR     GDDAT               ;1ST WORD
4431
4432 031266  010137  002420      3$:     MOV     R1,ADDR
4433 031272  023711  002422              CMP     GDDAT,(R1)          ;COMPARE
4434 031276  001431                      BEQ     4$                  ;GOOD BR
4435
4436 031300  011137  002424              MOV     (R1),BDDAT
4437
4438 031304  005737  002320              TST     FLAG
4439 031310  001007                      BNE     7$                  ;LOOK IF 1ST MESAGE OR EXTENDED ONE
4440 031312                              ERRHRD  51,EM0030,PRDMA
4441
4442 031322  005237  002320              INC     FLAG
4443 031326  000415                      BR      4$
4444
4445 031330                      7$:     ERRHRD  51,0,PDMAF          ;DATA CMP ERROR
4446 031340                              BREAK
4447 031342  005237  002320              INC     FLAG
4448 031346  022737  000010  002320      CMP     #10,FLAG            ;REPORT 10 FIRST ERROR
4449 031354  001002                      BNE     4$
4450 031356                              ESCAPE  SUB
4451
4452
4453
4454
4455 031362  005237  002422      4$:     INC     GDDAT               ;CHECK NEW LOCATION
4456 031366  062701  000002              ADD     #2,R1
4457 031372  022737  002000  002422      CMP     #2000,GDDAT         ;ALL DONE
4458 031400  001332                      BNE     3$
4459 031402                              ENDSUB
4460
4461
4462
4463
4464
```

```
4465 031404                          BGNSUB
4466 031406  005037 002320            CLR    FLAG
4467 031412  004737 014730            JSR    PC,MAINM1
4468 031416  012777 006426 161076     MOV    #RTABLE,@KMVP12      ;LOAD RX TABLE IN SEL12
4469 031424  005077 161070            CLR    @KMVP10
4470 031430  012777 002000 161066     MOV    #2000,@KMVP14        ;LOAD TABLE LENGTH
4471
4472
4473 031436  012702 002000           MOV    #2000,R2             ;TABLE LENGTH
4474 031442  012701 006426           MOV    #RTABLE,R1
4475 031446  005021           10$:   CLR    (R1)+                ;CLEAR RX TABLE
4476 031450  005302                   DEC    R2
4477 031452  001375                   BNE    10$
4478
4479
4480
4481
4482 031454  004537 015010            JSR    R5,TSTNUB            ;LOAD TEST NB21
4483 031460  000021                   .WORD  21
4484
4485 031462                           WAITB  0,1                  ;WAIT FOR TEST EXECUTION
4486
4487
4488 031502  004737 013152            JSR    PC,TSTERR            ;CHECK BSELO;WHICH ERROR
4489 031506  000423                   BR     2$                   ;TEST OK
4490 031510  000402                   BR     5$                   ;TIMEOUT ERROR
4491 031512  000411                   BR     6$                   ;NO ANSWER
4492 031514  000420                   BR     2$                   ;DATA CMP ERROR
4493
4494
4495
4496 031516  005037 002330    5$:     CLR    GOOD
4497 031522                           ERRHRD 52,EM0016,PBSELO     ;TIMEOUT ERROR
4498 031532                           ESCAPE SUB
4499
4500
4501
4502 031536  005037 002330    6$:     CLR    GOOD
4503 031542                           ERRHRD 53,EM0024,PBSELO     ;NO KMV11 ANSWER
4504 031552                           ESCAPE SUB
4505
4506
4507
4508
4509
4510
4511 031556  012702 002000    2$:     MOV    #2000,R2
4512 031562  012737 006426 002420     MOV    #RTABLE,ADDR         ;VERIFY RX TABLE
4513 031570  012737 006426 002422     MOV    #RTABLE,GDDAT
4514
4515 031576  023737 002422 002420 3$: CMP    GDDAT,ADDR           ;CMP TABLE
4516 031604  001432                   BEQ    4$
4517 031606  017737 150606 002424     MOV    @ADDR,BDDAT          ;READ BAD DATA
4518
4519
4520 031614  005737 002320            TST    FLAG
4521 031620  001007                   BNE    1$                   ;LOOK IF 1ST REPORT
```

```
4522
4523
4524 031622                                    ERRHRD  54,EM0030,PRDMA      ;DATA CMP ERROR IN RX TABLE
4525 031632  005237  002320                    INC     FLAG
4526 031636  000415                            BR      4$
4527
4528 031640                            1$:     ERRHRD  54,0,PDMAF           ;SHORT ERROR REPORT
4529 031650                                    BREAK
4530 031652  005237  002320                    INC     FLAG
4531 031656  022737  000010  002320            CMP     #10,FLAG             ;REPORT 10 ERROR
4532 031664  001002                            BNE     4$
4533 031666                                    ESCAPE  SUB
4534
4535
4536 031672  062737  000002  002422  4$:       ADD     #2,GDDAT             ;VERIFY NEXT LOCATION
4537 031700  062737  000002  002420            ADD     #2,ADDR
4538 031706  005302                            DEC     R2                   ;ALL DONE:
4539 031710  001332                            BNE     3$                   ;NO BRANCH
4540
4541 031712                            ENDSUB
4542 031714                            ENDTST
4543
```

```
KMV11 A/B LOGIC DIAG     MACRO M1200  06-JAN-83 09:39  PAGE 74
HARDWARE TESTS

    4545 031716                             BADHEAD
                                            ;******************* TEST21 *******************
    4546                                    ;TEST DMA TRANSFER IN BOTH DIRECTION
    4547 031716                             BADHEAD
                                            ;******************* TEST21 *******************
    4548
    4549
    4550
    4551
    4552
    4553 031716                             STARS 1
    4554                                     ;SET MAINT1    ;  DCT11 DECODE AND CLEAR BSEL0
    4555                                     ;THE HOST SET ALL THE PARAMETERS IN CSR'S
    4556                                     ;LOAD TX TABLE ADDRESS IN SEL12,TABLE LENGTH IN SEL14,EXTENDED ADDRESS IN BSEL10
    4557                                     ;EXTENDED ADDRESS OF RX TABLE IN BSEL2 , ADDRESS OF RX TABLE IN SEL4 AND
    4558                                     ;RAM STARTING ADDRESS FOR TRANSFER IN SEL6.
    4559                                     ;
    4560                                     ;
    4561                                     ;LOAD TEST NUMBER 22  ;  DCT11 EXECUTE TEST
    4562                                     ;  WHEN DONE CLEAR BSEL0 IF TEST OK OR SET 200 IN BSEL0 IF TIMEOUT DURING DMA.
    4563                                     ;
    4564                                     ;TEST DESCRIPTION:
    4565                                     ; HOST COMPUTER GENERATES DIFFERENT 1K WORD TABLES ,GIVES ALL PARAMETERS IN
    4566                                     ; THE CSR'S AND SET TEST 22 IN BSEL0
    4567                                     ; DCT11 TAKES SEL6 AS THE STARTING ADDRESS FOR THE DIFFERENT TRANSFERS IN KMV11
    4568                                     ; RAM MEMORY (DMA INTO KMV11) AND TRANSFER THIS TABLE IN DMA BACK TO HOST
    4569                                     ; MEMORY (DMA OUT).
    4570                                     ;
    4571                                     ; DATA TRANSFER ARE MADE IN DIFFERENT AREAS IN RAM AND  DCT11 CHECKS
    4572                                     ; THAT THE UNUSED PART OF THE RAM IS NOT MODIFIED
    4573                                     ;
    4574                                     ; WHEN TRANSFER IN BOTH DIRECTION HAS BEEN DONE ,DCT11 CLEAR BSEL0 AND
    4575                                     ; HOST COMPARES RX AND TX TABLE
    4576                                     ;
    4577                                     ;
    4578                                     ;ERROR REPORT IN BSEL0:          200=TIMEOUT DURING DMA
    4579                                     ;                                100=UNUSED MEMORY MODIFIED DURING TRANSFER
    4580                                     ;                                    IN THAT CASE SEL2 =GOOD
    4581                                     ;                                              SEL4 =BAD
    4582                                     ;                                              SEL6 = ADDRESS
    4583                                     ;
    4584                                     ;
    4585                                     ;
    4586 031716                             STARS 1
    4587
    4588
    4589
    4590
    4591 031716                             BGNTST
    4592 031716 012737 065000 012462            MOV    #65000,MAXCNT       ;RAM MEMORY MAX LENGTH
    4593 031724 005037 002320                   CLR    FLAG
    4594
    4595 031730 012703 000002                   MOV    #2,R3               ;SELECT 1ST PATTERN
    4596 031734 004737 014550         DMATWO:    JSR    PC,CLRKMV           ;CLEAR REG
    4597 031740 005037 002414                   CLR    COUNT               ;SELECT RAM STARTING ADDRESS FOR TX
    4598 031744                                 BREAK
    4599
```

```
4600 031746  004737  014750        TWODMA: JSR     PC,MAINM1              ;SET MAINT 1
4601
4602
4603
4604 031752  012702  002000                MOV     #2000,R2
4605 031756  012701  006426                MOV     #RTABLE,R1
4606 031762  005021                  10$:  CLR     (R1)+                  ;CLEAR RX TABLE
4607 031764  005302                        DEC     R2
4608 031766  001375                        BNE     10$
4609
4610
4611
4612
4613
4614
4615 031770  012702  002000                MOV     #2000,R2
4616 031774  012701  002426                MOV     #TTABLE,R1
4617 032000  004737  013230          1$:   JSR     PC,GENER               ;MAKE A PATTERN
4618
4619 032004  013721  012436                MOV     DATA,(R1)+             ;WRITE ONE TABLE LOCATION
4620 032010  005302                        DEC     R2                     ;ALL LOCATION DONE?
4621 032012  001372                        BNE     1$                     ;NO
4622
4623 032014  005077  160500                CLR     @KMVP10                ;CLEAR EXTENDED ADDRESS
4624 032020  013777  002414  160500        MOV     COUNT,@KMVP16          ;LOAD STATING ADDRESS IN RAM
4625
4626 032026  012777  002426  160466        MOV     #TTABLE,@KMVP12        ;SEND TX TABLE ADDRESS
4627 032034  012777  002000  160462        MOV     #2000,@KMVP14          ;SEND TABLE LENGTH
4628 032042  012777  006426  160444        MOV     #RTABLE,@KMVP04        ;SEND RX TABLE IN SEL4
4629 032050  005077  160436                CLR     @KMVP02                ;CLR RX TABLE EXT ADDRESS
4630 032054  004537  015010                JSR     R5,TSTNUB              ;LOAD TEST NB 22
4631 032060  000022                        .WORD   22
4632
4633 032062  012737  070000  002324        MOV     #70000,DELCT1          ; SET DELAY COUNTER
4634 032070  117700  160414          11$:  MOVB    @KMVCSR,R0             ; GET BSELO
4635 032074  105700                        TSTB    R0                     ; SEE IF TEST DONE OR CSR/DMA INTERFERENCE
4636 032076  001452                        BEQ     3$                     ; CHECK XMT/RCV BUFFER
4637 032100  005237  002324                INC     DELCT1                 ; UPDATE TIMEOUT COUNTER
4638 032104  001371                        BNE     11$                    ; BR IF NOT TIMED OUT
4639 032106  000407                        BR      6$                     ; TIME-OUT
4640
4641 032110                               BREAK
4642
4643
4644 032112  004737  013152                JSR     PC,TSTERR              ;CHECK BSELO;WHICH ERROR
4645 032116  000442                        BR      3$                     ;TEST OK
4646 032120  000402                        BR      6$                     ;TIME OUT
4647 032122  000411                        BR      7$                     ;NO KMV11 ANSWER
4648 032124  000420                        BR      20$                    ;PROBLEM IN THE UNUSED PART OF RAM:
4649                                                                       ;DMA TRANSFER MODIFY UNUSED RAM
4650                                                                       ; LOCATIONS.
4651
4652
4653
4654 032126  005037  002330          6$:   CLR     GOOD
4655 032132                                ERRHRD  55,EM0016,PBSELO       ;TIMEOUT ERROR
4656 032142                                ESCAPE  TST
```

```
4657
4658
4659
4660 032146  005037 002330        7$:     CLR      GOOD
4661 032152                               ERRHRD   56,EM0024,PBSELO      ;NO KMV11 ANSWER
4662 032162                               ESCAPE   TST
4663
4664
4665
4666 032166  017737 160324 002420 20$:    MOV      @KMVP06,ADDR         ;READ ADD IN ERROR
4667 032174  017737 160312 002330         MOV      @KMVP02,GOOD         ;GOOD VALUE
4668 032202  017737 160306 002424         MOV      @KMVP04,BDDAT        ;READ WRONG VALUE
4669 032210                               ERRHRD   57,EM0033,PDMARA     ;DATA  ERROR IN RAM DURING TRANSFER
4670 032220                               ESCAPE   TST
4671
4672
4673
4674
4675 032224  005077 160260        3$:     CLR      @KMVCSR
4676 032230  000240                        NOP
4677 032232  012777 044000 160250         MOV      #MAINT1,@KMVCSR      ;STOP TEST 22 IN KMV
4678 032240  012701 002426               MOV      #TTABLE,R1           ;TX TABLE ADDRESS
4679 032244  012704 006426               MOV      #RTABLE,R4           ;RX  ''     ''
4680 032250  012702 002000               MOV      #2000,R2             ;TABLE LENGHT
4681
4682
4683
4684 032254  021114                4$:     CMP      (R1),(R4)            ;CMP RX TABLE AND TX TABLE
4685 032256  001437                        BEQ      5$                   ;OK TEST NEXT LOCATION
4686
4687
4688 032260  011137 002422               MOV      (R1),GDDAT           ;PREPARE ERROR REPORT
4689 032264  011437 002424               MOV      (R4),BDDAT
4690 032270  010437 002420               MOV.     R4,ADDR
4691 032274  005037 012426               CLR      EXADDR
4692
4693 032300  005737 002320               TST      FLAG
4694 032304  001007                        BNE      2$
4695 032306                               ERRHRD   58,EM0021,PRDMA      ;DATA CMP ERROR IN TABLE
4696 032316  005237 002320               INC      FLAG
4697 032322  000415                        BR       5$
4698
4699
4700 032324                        2$:     ERRHRD   58,0,PDMAF           ;REPORT 10 FIRST ERROR
4701 032334                               BREAK
4702 032336  005237 002320               INC      FLAG
4703 032342  022737 000010 002320         CMP      #10,FLAG
4704 032350  001002                        BNE      5$
4705 032352                               ESCAPE   TST
4706
4707
4708
4709
4710 032356  005721                5$:     TST      (R1)+
4711 032360  005724                        TST      (R4)+
4712 032362  005302                        DEC      R2                   ;ALL MEMORY TESTED?
4713 032364  001333                        BNE      4$                   ;NO BRANCH
```

KMV11 A/B LOGIC DIAG   MACRO M1200 06-JAN-83 09:39 PAGE 74-3
HARDWARE TESTS

```
4714
4715
4716 032366  062737  002000  002414        ADD     #2000,COUNT        ;USE OTHER PART OF RAM
4717 032374  023737  002414  012462        CMP     COUNT,MAXCNT       ;IS ALL RAM USED?
4718 032402  100002                         BPL     30$
4719 032404  000137  031746                JMP     TWODMA
4720
4721
4722
4723 032410  005203                  30$:   INC     R3                 ;SELECT NEW KIND OF PATTERN
4724 032412  022703  000005                CMP     #5,R3              ;ALL DONE?
4725 032416  001402                         BEQ     40$                ;NO BRANCH
4726 032420  000137  031746                JMP     TWODMA
4727 032424                          40$:
4728 032424                          ENDTST
4729
```

KMV11 A/B LOGIC DIAG   MACRO M1200   06-JAN-83 09:39 PAGE 75
HARDWARE TESTS

```
   4731 032426                              BADHEAD
                                           ;******************** TEST22 ********************
   4732                                     ;TEST INTERUPT CAPABILITY OF KMV11 MODULE ON QBUS
   4733 032426                              BADHEAD
                                           ;******************** TEST22 ********************
   4734
   4735
   4736
   4737 032426                              STARS 1
   4738                                     ;SET MAINT1    ; KMV11 DECODE AND CLEAR BSELO
   4739                                     ;HOST PREPARES VECTOR AREA
   4740                                     ;SEND TEST NUMBER (23 OR 24)
   4741                                     ;DCT11 INTERUPTS THE HOST BY SETTING BITS 5 OR 6 IN ADDRESS 140000 OF
   4742                                     ;KMV11 MICRO BUS ;DCT11 CLEAR BSELO WHEN TEST COMPLETED.
   4743                                     ;
   4744                                     ;HOST TESTS IF THE INTERUPT HAS BEEN RECEIVED WITH CORRECT VECTOR
   4745                                     ;
   4746                                     ;
   4747                                     ;
   4748                                     ;
   4749                                     ;MICRO TEST 23 =INTERUPT ON LOW VECTOR
   4750                                     ;MICRO TEST 24 =INTERUPT ON HIGH VECTOR
   4751 032426                              STARS 1
   4752
   4753
   4754
   4755
   4756
   4757
   4758
   4759 032426                              BGNTST
   4760 032426  004737  014550                      JSR    PC,CLRKMV              ;CLR REG
   4761 032432  004737  014730                      JSR    PC,MAINM1
   4762 032436                              BGNSUB
   4763 032440  005037  012430                      CLR    INTFLG
   4764 032444  013702  012476                      MOV    KMVLVL,R2             ;READ KMV PRIORITY
   4765 032450  012777  032534  160016             MOV    #INT1,@KMVV00         ;SET UP VECTOR 0
   4766 032456  006202                              ASR    R2
   4767 032460  006202                              ASR    R2
   4768 032462  006202                              ASR    R2
   4769 032464  006202                              ASR    R2
   4770 032466  012777  000340  160006             MOV    #340,@KMVV02          ;SET KMV PRIORITY 7 FOR INTERUPT
   4771
   4772
   4773
   4774 032474  012703  000340                      MOV    #340,R3              ;TRY PRIORITY 7 FOR PROCESSOR
   4775
   4776
   4777 032500  106403                      SETPR1: MTPS   R3                    ;LOAD PRIORITY
   4778 032502  004537  015010                      JSR    R5,TSTNUB             ;SEND TEST 23
   4779 032506  000023                              .WORD  23
   4780
   4781 032510  000240                              NOP
   4782 032512  000240                              NOP
   4783 032514  000240                              NOP
   4784 032516  000240                              NOP
   4785 032520  000240                              NOP
```

KMV11 A/B LOGIC DIAG   MACRO M1200   06-JAN-83 09:39 PAGE 75-1
HARDWARE TESTS

```
4786 032522  000240                           NOP
4787 032524  000240                           NOP
4788
4789 032526                                   BREAK
4790 032530  000137  032544                   JMP     VECTO
4791
4792
4793                          ;***************INTERUPT  ROUTINE  ****************************
4794
4795 032534  052737  000001  012430  INT1:    BIS     #1,INTFLG               ;SET INT FLAG
4796 032542  000002                            RTI
4797
4798
4799                          ;***********************************************************************
4800
4801
4802
4803 032544  004537  013722         VECTO:    JSR     R5,CBSELO               ;CHECK IF KMV11 ANSWER
4804 032550  000000                            .WORD   0
4805 032552  000410                            BR      2$
4806 032554                                    ERRHRD  59,EM0024               ;NO KMV11 ANSWER
4807 032564  004737  013010                    JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
4808 032570                                    ESCAPE  SUB
4809
4810
4811 032574  005737  012430         2$:       TST     INTFLG                  ;TEST IF INTERUPT ?
4812 032600  001454                            BEQ     3$
4813
4814 032602  010237  002330                    MOV     R2,GOOD                 ;GOOD INTERUPT LEVEL
4815
4816
4817
4818 032606  062703  000040                    ADD     #40,R3                  ;WAS IT LEGAL
4819 032612  010337  012432                    MOV     R3,BAD
4820 032616  023737  012432  002330            CMP     BAD,GOOD
4821 032624  001461                            BEQ     4$                      ;YES BRANCH
4822
4823 032626  106237  002330                    ASRB    GOOD
4824 032632  106237  002330                    ASRB    GOOD
4825 032636  106237  002330                    ASRB    GOOD
4826 032642  106237  002330                    ASRB    GOOD
4827 032646  106237  002330                    ASRB    GOOD
4828 032652  042737  177770  002330            BIC     #177770,GOOD
4829
4830
4831 032660  106237  012432                    ASRB    BAD
4832 032664  106237  012432                    ASRB    BAD
4833 032670  106237  012432                    ASRB    BAD
4834 032674  106237  012432                    ASRB    BAD
4835 032700  106237  012432                    ASRB    BAD
4836 032704  042737  177770  012432            BIC     #177770,BAD
4837 032712                                    ERRHRD  60,EM0022
4838 032722  004737  013010                    JSR     PC,CHKMAX
4839 032726                                    ESCAPE  SUB
4840
4841
4842
```

```
KMV11 A/B LOGIC DIAG    MACRO M1200  06-JAN-83 09:39  PAGE 75-2
HARDWARE TESTS

4843 032732  122703  000140    3$:     CMPB    #140,R3              ;IS PROCESSOR AT LEVEL 3
4844 032736  001404                     BEQ     5$
4845 032740  162703  000040            SUB     #40,R3               ;DECREASE PRIORITY
4846 032744  000137  032500            JMP     SETPR1               ;TRY WITH NEW ONE
4847
4848
4849
4850 032750                    5$:     ERRHRD  61,EM0023            ;NO INTERUPT OCCUR
4851 032760  004737  013010            JSR     PC,CHKMAX            ;CHECK IF TOO MANY ERROR
4852 032764                            ESCAPE  SUB
4853
4854 032770                    4$:
4855 032770                    ENDSUB
4856
4857
4858 032772                    BGNSUB
4859 032774  004737  014730            JSR     PC,MAINM1
4860 033000  005037  012430            CLR     INTFLG
4861 033004  013702  012476            MOV     KMVLVL,R2            ;SET PRIORITY LEVEL
4862 033010  012777  000340  157466    MOV     #340,aKMVV06
4863 033016  012777  033070  157454    MOV     #INT2,aKMVV04        ;SET UP VECTOR 4
4864 033024  006202                    ASR     R2
4865 033026  006202                    ASR     R2
4866 033030  006202                    ASR     R2
4867 033032  006202                    ASR     R2
4868 033034  012703  000340            MOV     #340,R3              ;START WITH PRIORITY 7 FOR PROCESSOR
4869
4870
4871
4872
4873 033040  106403            INTPR2: MTPS    R3                   ;LOAD PRIORITY
4874 033042  004537  015010            JSR     R5,TSTNUB
4875 033046  000024                    .WORD   24                   ;SET TEST NB 24
4876 033050  000240                    NOP
4877 033052  000240                    NOP
4878 033054  000240                    NOP
4879 033056  000240                    NOP
4880 033060  000240                    NOP
4881
4882 033062                            BREAK
4883 033064  000137  033100            JMP     VECT4
4884
4885
4886                            ;****************INTERUPT ROUTINE ********************************
4887
4888
4889 033070  052737  000001  012430 INT2: BIS   #1,INTFLG            ;SET FLAG
4890 033076  000002                    RTI
4891
4892
4893                            ;**********************************************************************
4894
4895
4896
4897 033100  004537  013722    VECT4:  JSR     R5,CBSEL0            ;IS THERE KMV11 ANSWER ?
4898 033104  000000                    .WORD   0
4899 033106  000410                    BR      2$
```

```
4900 033110                               ERRHRD  62,EM0024        ;NO KMV11 ANSWER
4901 033120   004737  013010              JSR     PC,CHKMAX        ;CHECK IF TOO MANY ERROR
4902 033124                               ESCAPE  SUB
4903
4904
4905
4906 033130   032737  000001  012430  2$: BIT     #1,INTFLG        ;TEST IF INTERUPT OCCUR
4907 033136   001454                       BEQ     3$               ;NO INTERUPT
4908
4909 033140   010237  002330               MOV     R2,GOOD          ;GOOD INTERUPT LEVEL
4910
4911
4912 033144   062703  000040               ADD     #40,R3           ;WAS IT LEGAL ?
4913 033150   010337  012432               MOV     R3,BAD
4914
4915 033154   023737  012432  002330       CMP     BAD,GOOD
4916 033162   001461                        BEQ     4$               ;YES BRANCH
4917 033164   106237  002330                ASRB    GOOD
4918 033170   106237  002330                ASRB    GOOD
4919 033174   106237  002330                ASRB    GOOD
4920 033200   106237  002330                ASRB    GOOD
4921 033204   106237  002330                ASRB    GOOD
4922 033210   042737  177770  002330        BIC     #177770,GOOD     ;GET ACTUAL LEVEL
4923
4924 033216   106237  012432                ASRB    BAD
4925 033222   106237  012432                ASRB    BAD
4926 033226   106237  012432                ASRB    BAD
4927 033232   106237  012432                ASRB    BAD
4928 033236   106237  012432                ASRB    BAD
4929 033242   042737  177770  012432        BIC     #177770,BAD
4930 033250                                 ERRHRD  63,EM0022        ;INT OCCUR AT BAD LEVEL
4931 033260   004737  013010                JSR     PC,CHKMAX
4932 033264                                 ESCAPE  SUB
4933
4934
4935
4936
4937 033270   122703  000040            3$: CMPB    #40,R3           ;IS PROCESSOR AT PRIORITY 3
4938 033274   001404                        BEQ     5$               ;YES,NO INTERUPT OCCURED
4939 033276   162703  000040                SUB     #40,R3           ;DECRESE PRIORITY LEVEL
4940 033302   000137  033040                JMP     INTPR2
4941
4942
4943 033306                            5$:  ERRHRD  64,EM0023
4944 033316   004737  013010                JSR     PC,CHKMAX        ;CHECK IF TOO MANY ERROR
4945 033322                                 ESCAPE  SUB
4946 033326                            4$:
4947 033326                                 ENDSUB
4948 033330                                 ENDTST
```

```
4950
4951 033332                          BADHEAD
                                     ;********************** TEST23 ******************
                                     ;TEST INTERUPT ON DCT11 MICROPROCESSOR
4952                                 BADHEAD
4953 033332                          ;********************** TEST23 ******************
4954
4955
4956
4957
4958
4959
4960 033332                          STARS 1
4961                                 ;CHECKS THAT QBUS ACCESS ON BSEL0 AND BSEL10 CAUSE AN INTERUPT ON DCT11
4962                                 ;CHECKS THAT ACCESSES ON ALL THE OTHER CSR'S DOES NOT CAUSE ANY INTERUPTS.
4963                                 ;
4964                                 ;
4965                                 ;
4966                                 ;TEST DESCRIPTION:
4967                                 ;TEST NUMB 25:  DCT11 INITIALIZE VECTOR 60 ON DCT11 BUS CORRESPONDING TO
4968                                 ;               BSEL0 INTERUPT
4969                                 ;
4970                                 ;               QBUS ACCESS ALL REGISTERS BUT BSEL0  AND CHECK THAT NO
4971                                 ;               INTERUPT OCCUR ON DCT11
4972                                 ;
4973                                 ;
4974                                 ;
4975                                 ;
4976                                 ;               CHECK THAT QBUS ACCESS ON BSEL0 GIVE AN INTERUPT ON VECTOR 60
4977                                 ;
4978                                 ;
4979                                 ;
4980                                 ;TEST NUMB 26:  DCT11 INITIALIZE VECTOR 70 CORRESPONDING TO BSEL2
4981                                 ;               INTERUPT
4982                                 ;
4983                                 ;
4984                                 ;               QBUS ACCESS ALL REGISTERS BUT BSEL2 AND CHECK NO INTERUPT
4985                                 ;               OCCUR ON DCT11
4986                                 ;
4987                                 ;
4988                                 ;               CHECK THAT QBUS ACCES ON BSEL2 INTERUPT ON VECTOR 70
4989                                 ;
4990                                 ;
4991                                 ;ERROR REPORTING:        BSEL0=0         IF INTERUPT OCCUR
4992                                 ;                        BSEL0=100       IF ILLEGAL VECTOR
4993                                 ;                        BSEL0=TST NB    IF NO INTERUPT
4994                                 ;                        SEL2 =          EXPECTED VECTOR
4995 033332                          STARS 1
```

```
4997
4998
4999
5000
5001
5002
5003
5004
5005 033332                          BGNTST
5006
5007
5008 033332                          BGNSUB
5009 033334  004737 014550                    JSR     PC,CLRKMV        ;CLEAR REG
5010 033340  004737 014730                    JSR     PC,MAINM1        ;SET MAINT MODE
5011 033344  004537 015010                    JSR     R5,TSTNUB        ;SET TEST NB 25
5012 033350  000025                           .WORD   25
5013
5014
5015 033352                          WAITB   0,1
5016
5017
5018
5019 033372  013701 012512                    MOV     KMVP02,R1        ;LOAD CSR ADDR
5020 033376  012702 000012                    MOV     #12,R2           ;ACCES BSEL2 TO BSEL16
5021
5022 033402  152721 000207          1$:       BISB    #207,(R1)+       ;WRITE ALL REG BUT BSEL0
5023
5024 033406                          WAITB   0,1              ;WAIT FOR TEST EXECUTION
5025
5026 033426  004537 013722                    JSR     R5,CBSEL0        ;LOOK IF INTERUPT OCCUR
5027 033432  000000                           .WORD   0
5028
5029 033434  000404                           BR      3$               ;YES SEE WHICH ERROR
5030 033436  005302                           DEC     R2               ;ALL REG DONE ?
5031 033440  001360                           BNE     1$               ;NO BR
5032
5033
5034
5035
5036 033442  000137 033474                    JMP     GOON1            ;OK NO ACCESS INTERUPT THE DCT11 ;GO ON
5037
5038
5039
5040 033446  010137 002420          3$:       MOV     R1,ADDR          ;SEE WHICH ADDRESS CAUSE INTERUPT
5041 033452  162737 000001 002420             SUB     #1,ADDR
5042 033460                          ERRHRD  65,EM0026,PINTR  ;WRONG INTERUPT OCCURED ON DCT11
5043                                                         ;WHILE ADDRESSING KMV11 REGISTERS
5044 033470                          ESCAPE  SUB
5045
5046
5047
5048 033474  052777 004025 157006  GOON1:    BIS     #4025,@KMVCSR    ;ACCESS BSEL0
5049
5050 033502                          WAITB   0,1
5051
5052 033522  004537 013722                    JSR     R5,CBSEL0
5053 033526  000000                           .WORD   0
```

KMV11 A/B LOGIC DIAG   MACRO M1200   06-JAN-83 09:39   PAGE 77-1
HARDWARE TESTS

```
5054 033530  000424                             BR      5$              ;TEST OK ,INTRUPT OCCURED AT GOOD VECTOR
5055
5056 033532  004537  013722                     JSR     R5,CBSELO
5057 033536  000100                             .WORD   100
5058 033540  000410                             BR      6$              ;INT ON ILLEGAL VECTOR
5059 033542                                      ERRHRD  66,EM0027       ;NO KMV11 ANSWER, DCT11 DOES NOT RECEIVE ANY
5060 033552  004737  013010                      JSR     PC,CHKMAX       ;CHECK IF TOO MANY ERROR
5061 033556                                      ESCAPE  SUB             ;INTERUPT WHEN QBUS ADDRESS CSR'S
5062
5063
5064
5065
5066 033562                           6$:        ERRHRD  67,EM0032       ;INT ON ILLEGAL VECTOR WHEN ACCESSING BSELO
5067 033572  004737  013010                      JSR     PC,CHKMAX       ;CHECK IF TOO MANY ERROR
5068 033576                                      ESCAPE  SUB
5069
5070 033602  000240                   5$:        NOP
5071 033604                           ENDSUB
5072
5073
5074
5075
5076
5077 033606                           BGNSUB
5078 033610  004737  014730                      JSR     PC,MAINM1       ;SET MAINT MODE
5079 033614  004537  015010                      JSR     R5,TSTNUB       ;SET TEST NB 26
5080 033620  000026                              .WORD   26
5081
5082
5083 033622                                      WAITB   0,1
5084
5085 033642  052777  000026  156640             BIS     #26,@KMVCSR     ;WRITE SELO
5086
5087 033650  013701  012514                      MOV     KMVP04,R1       ;LOAD CSR ADDR
5088 033654  012702  000010                      MOV     #10,R2          ;ACCES BSEL3 TO BSEL11
5089
5090 033660  152721  000207           1$:        BISB    #207,(R1)+      ;WRITE ALL REG BUT BSEL2
5091
5092 033664                                      WAITB   0,1             ;WAIT FOR TEST EXECUTION
5093
5094 033704  004537  013722                      JSR     R5,CBSELO       ;LOOK IF INTERUPT OCCUR
5095 033710  000000                              .WORD   0
5096
5097 033712  000404                              BR      3$              ;YES SEE WHICH ERROR
5098 033714  005302                              DEC     R2              ;ALL REG DONE ?
5099 033716  001360                              BNE     1$              ;NO BR
5100
5101
5102
5103
5104
5105 033720  000137  033756                      JMP     GOON2           ;OK NO ACCESS INTERUPT THE DCT11 ;GO ON
5106
5107
5108 033724  017737  156564  012440   3$:        MOV     @KMVP04,VECT    ;READ RECEIVE VECTOR
5109 033732  010137  002420                      MOV     R1,ADDR         ;SEE WHICH ADDRESS MAKE INTERUPT
5110 033736  005337  002420                      DEC     ADDR
```

```
5111 033742                                 ERRHRD  68,EM0026,PINTR ;WRONG INTERUPT OCCUR WHILE ACCESSING REGISTERS
5112 033752                                 ESCAPE  SUB
5113
5114
5115 033756  000240            GOON2:       NOP
5116
5117 033760  052777  017777  156524         BIS     #17777,@KMVP02  ;ACCESS BSEL2
5118
5119 033766                                 WAITB   0,1
5120
5121 034006  004537  013722                 JSR     R5,CBSELO
5122 034012  000000                          .WORD   0
5123 034014  000424                          BR      5$              ;TEST OK ,INTRUPT OCCUR AT GOOD VECTOR
5124
5125 034016  004537  013722                 JSR     R5,CBSELO
5126 034022  000100                          .WORD   100
5127 034024  000410                          BR      6$              ;INT ON ILLEGAL VECTOR
5128 034026                                 ERRHRD  69,EM0027       ;NO KMV11 ANSWER
5129 034036  004737  013010                 JSR     PC,CHKMAX       ;CHECK IF TOO MANY ERROR
5130 034042                                 ESCAPE  SUB
5131
5132
5133
5134
5135
5136 034046            6$:                  ERRHRD  70,EM0026               ;INT ON ILLEGAL VECTOR
5137 034056  004737  013010                 JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
5138 034062                                 ESCAPE  SUB
5139
5140 034066  000240            5$:          NOP
5141 034070            ENDSUB
5142
5143
5144
5145 034072            ENDTST
5146
```

```
KMV11 A/B LOGIC DIAG     MACRO M1200   06-JAN-83 09:39  PAGE 78
HARDWARE TESTS

5148
5149                                    .SBTTL HARDWARE PARAMETER CODING SECTION
5150
5151
5152
5153                                    ;////////////////////////////////////////////////////////////////////////
5154                                    ;/ THE HARDWARE PARAMETER CODING SECTION CONTAINS MACROS
5155                                    ;/ THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
5156                                    ;/ MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
5157                                    ;/ INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
5158                                    ;/ MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
5159                                    ;/ WITH THE OPERATOR.
5160                                    ;////////////////////////////////////////////////////////////////////;////
5161
5162 034074                             BGNHRD
5163
5164 034076                             GPRMA     ADDRES,0,0,60000,177776,YES
5165 034106                             GPRMA     VECTOR,2,0,0,674,YES
5166 034116                             GPRMD     PRIRTY,4,0,7000,4,7,YES
5167 034130                             ENDHRD
5168
5175
5176
5177 034130       115      111      103 ADDRES: .ASCIZ  /MICRO-CPU  CSR ADDRESS : /
     034133       122      117      055
     034136       103      120      125
     034141       040      040      103
     034144       123      122      040
     034147       101      104      104
     034152       122      105      123
     034155       123      040      072
     034160       040      000
5178 034162       115      111      103 VECTOR: .ASCIZ  /MICRO-CPU VECTOR ADDRESS : /
     034165       122      117      055
     034170       103      120      125
     034173       040      126      105
     034176       103      124      117
     034201       122      040      101
     034204       104      104      122
     034207       105      123      123
     034212       040      072      040
     034215       000
5179 034216       115      111      103 PRIRTY: .ASCIZ  /MICRO-CPU PRIORITY LEVEL : /
     034221       122      117      055
     034224       103      120      125
     034227       040      120      122
     034232       111      117      122
     034235       111      124      131
     034240       040      114      105
     034243       126      105      114
     034246       040      072      040
     034251       000
5180
5181                                    .EVEN
5182
5183
5184
```

KMV11 A/B LOGIC DIAG   MACRO M1200   06-JAN-83 09:39   PAGE 78-1
HAFDWARE PARAMETER CODING SECTION

       5185
       5186
       5187

I 10

SEG 125

KMV11 A/B LOGIC DIAG   MACRO M1200   06-JAN-83 09:39 PAGE 79
SOFTWARE PARAMETER CODING SECTION

```
5189                          .SBTTL  SOFTWARE PARAMETER CODING SECTION
5190
5191
5192                  ;////////;////,,//////////////////////////////////////////////////////
5193                  ;/ THE SOFTWARE PARAMETER CODING SECTION CONTAINS MACROS
5194                  ;/ THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.   THE
5195                  ;/ MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
5196                  ;/ INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.   THE
5197                  ;/ MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
5198                  ;/ WITH THE OPERATOR.
5199                  ;///////////////////////////////////////////////////////////////////////
5200
5201 034252                  BGNSFT
5202
5211
5212
5213 034254                  ENDSFT
5214
5215
5222
5223
```

```
5225
5226 034254                        $PATCH::
5227 034254                               .BLKW   50
5228
5235
5236 034374                        LASTAD
     034400                        L$LAST::
5237 034400                               ENDMOD
5238
5239
```

```
5241
5242
5255
5256 034400                            BGNSETUP        1
5257 034400                            BGNPTAB
5258 034404  177000                    .WORD   177000
5259 034406  000300                    .WORD   300
5260 034410  004000                    .WORD   4000
5261 034412  000001                    .WORD   1
5262 034414                            ENDPTAB
5263 034414                            ENDSETUP
5264
5265
5266
5267
5268
5269         000001            .END
```

KMV11 A/B LOGIC DIAG   MACRO M1200   06-JAN-83 09:39   PAGE 81-1
SYMBOL TABLE

Symbol	Value		Symbol	Value		Symbol	Value		Symbol	Value		Symbol	Value	
ABORT	024010		C$DCLN=	000044		EF.RES=	000037	G	F$RPT =	000012		G$RADL=	000120	
ADDR	002420		C$DODU=	000051		EF.STA=	000040	G	F$SEG =	000003		G$RADO=	000020	
ADDRES	034130		C$DRPT=	000024		EM0001	015265		F$SOFT=	000005		G$XFER=	000004	
ADR =	000020	G	C$DU =	000053		EM0002	015361		F$SRV =	000010		G$YES =	000010	
ASSEMB=	000010		C$EDIT=	000003		EM0003	015424		F$SUB =	000002		HELP =	000000	
BAD	012432		C$ERDF=	000055		EM0004	015512		F$SW =	000014		HOE =	100000	G
BDDAT	002424		C$ERHR=	000056		EM0005	015575		F$TEST=	000001		IBE =	010000	G
BIT0 =	000001	G	C$ERRO=	000060		EM0006	015671		GDDAT	002422		IDU =	000000	G
BIT00 =	000001	G	C$ERSF=	000054		EM0007	015764		GDREV	012464		IER =	020000	G
BIT01 =	000002	G	C$ERSO=	000057		EM0010	016056		GENER	013230		INIFLG	012466	
BIT02 =	000004	G	C$ESCA=	000010		EM0011	016152		GENER1	013352		INTFLG	012430	
BIT03 =	000010	G	C$ESEG=	000005		EM0012	016246		GENEX	013510		INTPR2	033040	
BIT04 =	000020	G	C$ESUB=	000003		EM0013	016342		GENE1	025502		INT1	032534	
BIT05 =	000040	G	C$ETST=	000001		EM0015	016436		GENE2	025700		INT2	033070	
BIT06 =	000100	G	C$EXIT=	000032		EM0016	016506		GENE3	026076		ISR =	000100	G
BIT07 =	000200	G	C$GETB=	000026		EM0020	016544		GENE4	026306		IXE =	004000	G
BIT08 =	000400	G	C$GETW=	000027		EM0021	016630		GENE5	026510		I$AU =	000041	
BIT09 =	001000	G	C$GMAN=	000043		EM0022	016722		GENE6	026712		I$AUTO=	000041	
BIT1 =	000002	G	C$GPHR=	000042		EM0023	016762		GENINC	013502		I$CLN =	000041	
BIT10 =	002000	G	C$GPLO=	000030		EM0024	017005		GENISH	013360		I$DU =	000041	
BIT11 =	004000	G	C$GPRI=	000040		EM0025	017075		GENRAN	013362		I$HRD =	000041	
BIT12 =	010000	G	C$INIT=	000011		EM0026	017136		GENROT	013336		I$INIT=	000041	
BIT13 =	020000	G	C$INLP=	000020		EM0027	017232		GENR0	013324		I$MOD =	000041	
BIT14 =	040000	G	C$MANI=	000050		EM0028	017307		GENR1	013314		I$MSG =	000041	
BIT15 =	100000	G	C$MEM =	000031		EM0030	017341		GENSEL	013246		I$PROT=	000040	
BIT2 =	000004	G	C$MSG =	000023		EM0031	017423		GEN0	013266		I$PTAB=	000041	
BIT3 =	000010	G	C$OPEN=	000034		EM0032	017477		GEN1	013272		I$PWR =	000041	
BIT4 =	000020	G	C$PNTB=	000014		EM0033	017573		GEN25	013306		I$RPT =	000041	
BIT5 =	000040	G	C$PNTF=	000017		EM0034	017654		GEN52	013300		I$SEG =	000041	
BIT6 =	000100	G	C$PNTS=	000016		EM0035	020013		GETPRM	023606		I$SETU=	000041	
BIT7 =	000200	G	C$PNTX=	000015		EM0134	017746		GOOD	002330		I$SFT =	000041	
BIT8 =	000400	G	C$QIO =	000377		END	024100		GOOD0	002332		I$SRV =	000041	
BIT9 =	001000	G	C$RDBU=	000007		ERRBLK	002264	G	GOOD1	002334		I$SUB =	000041	
BOE =	000400	G	C$REFG=	000047		ERRCNT	002300		GOOD10	002344		I$TST =	000041	
BSEL0	012434		C$RESE=	000033		ERRMSG	002262	G	GOOD12	002346		J$JMP =	000167	
BSEL1	002376		C$REVI=	000003		ERRNBR	002260	G	GOOD14	002350		KIND	012442	
CBSEL0	013722		C$RFLA=	000021		ERRTYP	002256	G	GOOD16	002352		KMTLVL	012506	
CHANEL	012444		C$RPT =	000025		EVL =	000004	G	GOOD2	002336		KMVCSR	012510	
CHECK	024650		C$SEFG=	000046		EXADDR	012426		GOOD4	002340		KMVLVL	012476	
CHECK1	025014		C$SPRI=	000041		E$END =	002100		GOOD6	002342		KMVP02	012512	
CHKMAX	013010		C$SVEC=	000037		E$LOAD=	000035		GOON1	033474		KMVP04	012514	
CKALL	013760		C$TPRI=	000013		FLAG	002320		GOON2	033756		KMVP06	012516	
CKREG	014262		DATA	012436		FTIME	002316		G$CNT0=	000200		KMVP10	012520	
CKSEL0	013670		DATA1 =	052525	G	F$AU =	000015		G$DELM=	000372		KMVP12	012522	
CLRKMV	014550		DATA2 =	125252	G	F$AUTO=	000020		G$DISP=	000003		KMVP14	012524	
COUNT	002414		DELCT1	002324		F$BGN =	000040		G$EXCP=	000400		KMVP16	012526	
C$AU =	000052		DELCT2	002326		F$CLEA=	000007		G$HILI=	000002		KMVV00	012474	
C$AUTO=	000061		DFPTBL	002212	G	F$DU =	000016		G$LOLI=	000001		KMVV02	012502	
C$BRK =	000022		DH1	002322		F$END =	000041		G$NO =	000000		KMVV04	012500	
C$BSEG=	000004		DIAGMC=	000000		F$HARD=	000004		G$OFFS=	000400		KMVV06	012504	
C$BSUB=	000002		DMAIN	030442		F$HW =	000013		G$OFSI=	000376		KMV11A	002000	G
C$CEFG=	000045		DMAOUT	031102		F$INIT=	000006		G$PRMA=	000001		LENGTH	012454	
C$CLCK=	000062		DMATWO	031734		F$JMP =	000050		G$PRMD=	000002		LOCK	002274	
C$CLEA=	000012		DROPD	024226		F$MOD =	000000		G$PRML=	000000		LOE =	040000	G
C$CLOS=	000035		EF.CON=	000036	G	F$MSG =	000011		G$RADA=	000140		LOGDEV	002302	
C$CLP1=	000006		EF.NEW=	000035	G	F$PROT=	000021		G$RADB=	000000		LOKFLG	012470	
C$CVEC=	000036		EF.PWR=	000034	G	F$PWR =	000017		G$RADD=	000040		LOOP	012530	

KMV11 A/B LOGIC DIAG   MACRO M1200   06-JAN-83 09:39   PAGE 81-2
SYMBOL TABLE

LOT	= 000010 G	L10002	022056	L10073	031712	O$SETU= 000001	SAVSP	002306
L$ACP	002110 G	L10003	022144	L10074	032424	PADFLT 022346 G	SAVSTA	002412
L$APT	002036 G	L10004	022206	L10075	033330	PBSEL0 023174 G	SEL0	002354
L$AU	024260 G	L10005	022312	L10076	032770	PCHECK 022314 G	SEL1	002356
L$AUT	002070 G	L10006	022344	L10077	033326	PDMAF 023354 G	SEL10	002366
L$AUTO	024102 G	L10007	022402	L10100	034072	PDMARA 022210 G	SEL12	002370
L$CCP	002106 G	L10010	022712	L10101	033604	PINTR 023232 G	SEL14	002372
L$CLEA	024174 G	L10011	023172	L10102	034070	PNT = 001000 G	SEL16	002374
L$CO	002032 G	L10012	023230	L10103	034130	PRALL 022404 G	SEL2	002360
L$DEPO	002011 G	L10013	023262	L10104	034254	PRBYTE 022146 G	SEL4	002362
L$DESC	002222 G	L10014	023352	L10105	034404	PRDMA 023264 G	SEL6	002364
L$DESP	002076 G	L10015	023410	L10107	034414	PRI = 002000 G	SETPR1	032500
L$DEVP	002060 G	L10016	023416	MAINM0	014660	PRIRTY 034216	SETUP	023524
L$DISP	002132 G	L10017	024100	MAINM1	014730	PRI00 = 000000 G	SSTACK	012732
L$DLY	002116 G	L10020	024172	MAINT0= 054000 G	PRI01 = 000040 G	SVCGBL= 000000
L$DTP	002040 G	L10021	024176	MAINT1= 044000 G	PRI02 = 000100 G	SVCINS= 177777
L$DTYP	002034 G	L10022	024256	MAXCNT	012462	PRI03 = 000140 G	SVCSUB= 177777
L$DU	024200 G	L10023	024260	MAXERR	002276	PRI04 = 000200 G	SVCTAG= 177777
L$DUT	002072 G	L10024	024376	MAXPRI= 000340 G	PRI05 = 000240 G	SVCTST= 177777
L$DVTY	012732 G	L10025	024572	MBSEL0	021103	PRI06 = 000300 G	S$LSYM= 010000
L$EF	002052 G	L10026	024514	MBYTE	021371	PRI07 = 000340 G	TBYTE	025102
L$ENVI	002044 G	L10027	024570	MCHECK	021666	PROMCK 030324	TCSR	024766
L$ERRT	002256 G	L10030	024732	MCLR = 040000 G	PRRAM 022060 G	TCSRNB	024756
L$ETP	002102 G	L10031	025074	MDMAF	021737	PRREG 022714 G	TFM36	015164
L$EXP1	002046 G	L10032	025226	MDMAR1	021453	PRSEL0 022022 G	TGENE1	025472
L$EXP4	002064 G	L10033	025454	MDMAR2	021542	PSTACK 002304	TGENE2	025670
L$EXP5	002066 G	L10034	025342	MDMAR3	021610	QV.FLG 012471	TGENE3	026066
L$HARD	034076 G	L10035	025452	MDMA1	021240	RAMADD 027724	TGENE4	026276
L$HIME	002120 G	L10036	025650	MDMA2	021323	RAMCAD 030076	TGENE5	026500
L$HPCP	002016 G	L10037	025544	MINT	021051	RAMPAT 027504	TGENE6	026702
L$HPTP	002022 G	L10040	025646	MINTR	021145	RANCLC 013462	TIM	015247
L$HW	002212 G	L10041	026046	MRAM1	020707	RANDN 002406	TSELA	024626
L$ICP	002104 G	L10042	025742	MRAM2	020771	RANEX 013500	TSELB	024632
L$INIT	023420 G	L10043	026044	MREG0	020107	RANGEN 013402	TSEL10	026176
L$LADP	002026 G	L10044	026256	MREG10	020323	RANMTA 002404	TSEL12	026400
L$LAST	034400 G	L10045	026152	MREG12	020366	RANSEC 013466	TSEL14	026602
L$LOAD	002100 G	L10046	026254	MREG14	020431	RANSEL 002402	TSEL16	027004
L$LUN	002074 G	L10047	026460	MREG16	020474	RANST 002400	TSEL4	025570
L$MREV	002050 G	L10050	026354	MREG2	020152	RAN1 013414	TSEL6	025766
L$NAME	002000 G	L10051	G26456	MREG4	020215	RAN2 013432	TSPEED	012452
L$PRIO	002042 G	L10052	026662	MREG6	020260	RAN4 013470	TSTERR	013152
L$PROT	002122 G	L10053	026556	MSEL0	020042	READ 015072	TSTNUB	015010
L$PRT	002112 G	L10054	026660	MSEL10	020645	REGADR 012532	TSTREG	024574
L$REPP	002062 G	L10055	027064	MSFL2	020537	RESTST 024402	TTABLE	002426
L$REV	002010 G	L10056	026760	MSEL4	020602	REVPRO 030260	TWODMA	031746
L$RPT	023412 G	L10057	027062	NERRS	013102	RGALL 027110	TXDATA	012446
L$SOFT	034254 G	L10060	027502	NEXT	023532	ROMMAP 024262	T$ARGC= 000002
L$SPC	002056 G	L10061	027722	NUB	012456	RSTREG 013610	T$CODE= 002032
L$SPCP	002020 G	L10062	030074	NUMBER	002416	RTABLE 006426	T$ERRN= 000106
L$SPTP	002024 G	L10063	030246	O$APTS= 000000	RUNNIN 024016	T$EXCP= 000000
L$STA	002030 G	L10064	030312	O$AU = 000000	RXCNT 012460	T$FLAG= 000040
L$SW	002266	L10065	030440	O$BGNR= 000000	RXDATA 012450	T$FREE= 034414
L$TEST	002114 G	L10066	031076	O$BGNS= 000000	SAVE4 002312	T$GMAN= 000000
L$TIML	002014 G	L10067	030662	O$DU = 000001	SAVE6 002314	T$HILI= 000007
L$UIT	002270	L10070	031074	O$ERRT= 000000	SAVPC 002310	T$LAST= 000001
L$UNIT	002012 G	L10071	031714	O$GNSW= 000001	SAVPC1 002410	T$LOLI= 000004
L10001	002222	L10072	031402	O$POIN= 000001	SAVREG 013530	T$LSYM= 010000

KMV11 A/B LOGIC DIAG   MACRO M1200 06-JAN-83 09:39 PAGE 81-3
SYMBOL TABLE

T$LTNO= 000027	T$$AUT= 010020	T11.1	026462	T21	031716 G	T9.1	026050
T$NEST= 177777	T$$CLE= 010021	T11.2	026560	T22	032426 G	T9.2	026154
T$NS0 = 000000	T$$DAT= 010107	T12	026664 G	T22.1	032436	UAM = 000200 G
T$NS1 = 000005	T$$DU = 010022	T12.1	026664	T22.2	032772	UNIT	002272
T$NS2 = 000002	T$$HAR= 010103	T12.2	026762	T23	033332 G	UUT	012472
T$PCNT= 000000	T$$HW = 010001	T13	027066 G	T23.1	033332	VECT	012440
T$PTAB= 010106	T$$INI= 010017	T14	027504 G	T23.2	033606	VECTOR	034162
T$PTHV= 000001	T$$MSG= 010015	T15	027724 G	T3	024574 G	VECTO	032544
T$PTNU= 000001	T$$PC = 000001	T16	030076 G	T4	024734 G	VECT4	033100
T$SAVL= 177777	T$$PRO= 010000	T17	030250 G	T5	025076 G	WAIT1	013000
T$SEGL= 177777	T$$PTA= 010106	T18	030314 G	T6	025230 G	WAIT2	012760
T$SIZE= 0000C6	T$$RPT= 010016	T19	030442 G	T6.1	025240	WRITE	015044
T$SUBN= 000002	T$$SOF= 010104	T19.1	030452	T6.2	025344	X$ALWA= 000000
T$TAGL= 177777	T$$SUB= 010102	T19.2	030664	T7	025456 G	X$FALS= 000040
T$TAGN= 010110	T$$TES= 010100	T2	024400 G	T7.1	025456	X$OFFS= 000400
T$TEMP= 000000	T1	024262 G	T2.1	024400	T7.2	025546	X$TRUE= 000020
T$TEST= 000027	T10	026260 G	T2.2	024516	T8	025652 G	$LSTIN= 000000
T$TSTM= 177777	T10.1	026260	T20	031100 G	T8.1	025652	$LSTTA= 000000
T$TSTS= 000001	T10.2	026356	T20.1	031100	T8.2	025744	$PATCH	034254 G
T$$AU = 010023	T11	026462 G	T20.2	031404	T9	026050 G		

. ABS. 034414   000
         000000   001
ERRORS DETECTED: 0

VIRTUAL MEMORY USED: 29088 WORDS (114 PAGES)
DYNAMIC MEMORY: 21924 WORDS (84 PAGES)
ELAPSED TIME: 00:29:54
VKMAA0.BIN,VKMAA0=[64,3]LIBA.MLB/ML,[64,5]VKMAA0