

DMV11  
M8053 M8064

DMV11 MCTRL DIAG# 1  
CVDMAAO

AH-F262A-MC  
FICHE 1 OF 2

MAY 1981  
COPYRIGHT © 1981  
MADE IN USA



A large grid of technical data, likely a control matrix or diagnostic chart, consisting of numerous small tables or diagrams arranged in rows and columns. The content is dense and difficult to read due to the small size and high contrast of the image. The grid appears to be organized into several major sections or columns, each containing multiple rows of data points, possibly representing different control parameters or diagnostic steps. The overall layout is highly structured and repetitive, typical of a technical manual or control system documentation.



DMV11  
M8053 M8064

DMV11 MCTRL DIAG# 1  
CVDMAAO

AH-F262A MC  
FICHE 2 OF 2

MAY 1981  
COPYRIGHT © 1981  
MADE IN USA





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

.TITLE CVDMAAO DMV11 MCTRL DIAG #1  
.SBTTL PROGRAM DOCUMENT  
.REM ^

I D E N T I F I C A T I O N

PRODUCT CODE: AC-F261A-MC  
PRODUCT NAME: CVDMAAO DMV-11 MICRO-CONTROLLER STATIC DIAGNOSTIC PART 1  
PRODUCT DATE: JANUARY 1981  
MAINTAINER: DIAGNOSTICS MERRIMACK CC:38P  
AUTHORS: CHRIS BRIENEN  
RAY MARSHALL  
PURPOSE: THIS DIAGNOSTIC IS DESIGNED TO PERFORM STATIC LOGIC TESTS FOR  
THE M8053 OR M8064 (HEREAFTER REFERRED TO AS THE DMV OR DMV-11)

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT  
NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL  
EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO  
RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS DOCUMENT.

NO RESPONSIBILITY IS ASSUMED FOR THE USE OR RELIABILITY OF  
SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL OR ITS  
AFFILIATED COMPANIES.

COPYRIGHT (C) 1981 BY DIGITAL EQUIPMENT CORPORATION

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL  
DEC

PDP  
DECUS

UNIBUS  
DECTAPE

MASSBUS



CVDMAA.P11

12-DEC-80 15:59

PROGRAM DOCUMENT

42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55

HISTORY

REV

DATE

REASON

0

14-JAN-81

INITIAL RELEASE



CVDMAA.P11

12-DEC-80 15:59

PROGRAM DOCUMENT

CONTENTS56  
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  
93  
94  
95  
96  
97  
98  
99  
100  
101  
102

- 1.0 INTRODUCTION
- 2.0 HARDWARE REQUIREMENTS
- 3.0 PRELIMINARY PROGRAM REQUIREMENTS
- 4.0 GENERAL PROGRAM CONSIDERATIONS
  - 4.1 DIAGNOSTIC SUPERVISOR
  - 4.2 EXECUTION TIME
  - 4.3 XXDP+
  - 4.4 ACT/SLIDE
  - 4.5 APT
  - 4.6 MEMORY MANAGEMENT
  - 4.7 ERROR LOGGING
- 5.0 PROGRAM LOAD MEDIA
- 6.0 OPERATING INSTRUCTIONS
  - 6.1 LOADING AND STARTING PROCEDURES
    - 6.1.1 LOADING PROCEDURES
    - 6.1.2 STARTING PROCEDURES
    - 6.1.3 \*\* STEPS FOR QUICK AND SIMPLE EXECUTION \*\*
  - 6.2 INITIAL DIALOGUE
  - 6.3 PROGRAM OPTIONS
    - 6.3.1 START COMMAND
    - 6.3.2 RESTART COMMAND
    - 6.3.3 CONTINUE COMMAND
    - 6.3.4 PROCEED COMMAND
    - 6.3.5 ADD COMMAND
    - 6.3.6 DROP COMMAND
    - 6.3.7 PRINT COMMAND
    - 6.3.8 DISPLAY COMMAND
    - 6.3.9 FLAGS COMMAND
    - 6.3.10 ZFLAGS COMMAND
    - 6.3.11 CONTROL CHARACTERS
    - 6.3.12 HARDWARE PARAMETERS
    - 6.3.13 SOFTWARE PARAMETERS
    - 6.3.14 EXTENDED DISCUSSION OF P-TABLE DIALOGUE
- 7.0 TEST DESCRIPTIONS
- 8.0 ERROR INFORMATION
  - 8.1 ERROR REPORTING



CVDMAA.P11

12-DEC-80 15:59

## PROGRAM DOCUMENT

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  
151  
152  
153  
154  
155  
156  
157  
158

## 1.0 INTRODUCTION

THE M8053 AND M8064 ARE SINGLE-LINE SYNCHRONOUS, MICRO-PROCESSOR BASED COMMUNICATIONS INTERFACES WHICH CAN SUPPORT BOTH CHARACTER-ORIENTED (DDCMP, BSC, ETC.) AND BIT-ORIENTED (SDLC, HDLC, ETC.) PROTOCOLS. THE PURPOSE OF THIS PROGRAM IS TO PERFORM DIAGNOSTIC TESTING OF THE CSRS, RAM, AND BASIC MICRO-PROCESSOR LOGIC ON THESE BOARDS. THE FOLLOWING FUNCTIONS WILL BE PERFORMED: DMV RESIDENT U-DIAG EXECUTION CSR ADDRESSING, VIA REGISTER STATIC BIT INTERACTION AND READ/WRITE TESTING, AND ON-BOARD RAM TESTING.

THE STATIC LOGIC TESTS WILL PROVIDE EXTENSIVE TROUBLESHOOTING CAPABILITIES, SUCH AS TIGHT SCOPE LOOPS, SWITCH OPTIONS, AND ABILITY TO 'LOCK' ONTO INTERMITTENT ERRORS. IN ADDITION TESTS ARE DESIGNED AND STRUCTURED TO ACHIEVE MAXIMUM FAULT RESOLUTION AND FACILITATE REPLACEMENT OF THE SMALLEST FIELD REPLACEABLE UNIT.

THIS PROGRAM IS IMPLEMENTED USING THE DIAGNOSTIC SUPERVISOR AND A STRUCTURED PROGRAMMING APPROACH. BECAUSE THE DESIGN CONFORMS TO THE SUPERVISOR (STANDALONE VERSION) THE PROGRAM IS COMPATIBLE WITH ACT, APT, XXDP+, AND SLIDE.

THROUGH DIALOGUE WITH THE OPERATOR, THE PROGRAM ALLOWS MODIFICATION OF DEVICE PARAMETERS, SUCH AS LSI-BUS ADDRESS, VECTOR ADDRESSES AND DEVICE PRIORITY. IN ADDITION, THE OPERATOR CAN SPECIFY PARTICULAR TESTS TO BE RUN AND A VARIETY OF LOOPING, RUNNING, AND REPORTING MODES.

DEVICE ERRORS WILL BE REPORTED AS THEY OCCUR. THE REPORT WILL INCLUDE A TEST NUMBER AND DESCRIPTION OF THE ERROR, GOOD AND BAD TEST DATA, AND APPLICABLE DEVICE REGISTER CONTENTS.

## 2.0 HARDWARE REQUIREMENTS

THE FOLLOWING HARDWARE IS REQUIRED TO RUN THE M8053/8064 STATIC LOGIC TESTS:

PDP-11/03 OR PDP-11/23  
16K WORDS OF MEMORY  
CONSOLE TERMINAL  
M8053 OR M8064 COMMUNICATIONS INTERFACE

## 3.0 PRELIMINARY PROGRAM REQUIREMENTS

THIS PROGRAM (CVDMA) SHOULD BE THE FIRST OF THE FIVE DMV-11 STATIC DIAGNOSTICS TO BE RUN. ERRORS FOUND IN THIS PROGRAM SHOULD BE CORRECTED BEFORE RUNNING THE OTHERS.



CVDMAA.P11

12-DEC-80 15:59

## PROGRAM DOCUMENT

159  
160  
161  
162  
163  
164  
165  
166  
167  
168  
169  
170  
171  
172  
173  
174  
175  
176  
177  
178  
179  
180  
181  
182  
183  
184  
185  
186  
187  
188  
189  
190  
191  
192  
193  
194  
195  
196  
197  
198  
199  
200  
201  
202  
203  
204  
205  
206  
207  
208  
209  
210  
211  
212  
213  
214

## 4.0 GENERAL PROGRAM CONSIDERATIONS

## 4.1 DIAGNOSTIC SUPERVISOR

THIS PROGRAM IS COMPATIBLE WITH THE STANDALONE DIAGNOSTIC SUPERVISOR, AND MUST BE LOADED TO BE CO-RESIDENT WITH THE SUPERVISOR, OR BE PREVIOUSLY COMBINED WITH THE SUPERVISOR AND LOADED AS A SINGLE FILE. IN EITHER CASE, THE COMBINED PROGRAM WILL NOT EXCEED 16K OF MEMORY.

## 4.2 EXECUTION TIME

THE MAXIMUM TIME REQUIRED TO RUN THIS PROGRAM PER PASS FOR EACH UNIT IS AS FOLLOWS: 11/03 = 100 SEC, 11/23 = 50 SECONDS.

## 4.3 XXDP+

THIS PROGRAM MAY BE LOADED UNDER XXDP+, AND MAY BE RUN IN DUMP MODE OR CHAIN MODE.

## 4.4 ACT/SLIDE

THIS PROGRAM MAY BE LOADED UNDER ACT OR SLIDE AND MAY BE RUN IN DUMP MODE OR CHAIN MODE.

## 4.5 APT

THIS PROGRAM MAY BE LOADED BY THE APT SYSTEM (INCLUDING APT-RD) AND RUN IN PROGRAM MODE OR SCRIPT MODE.

## 4.6 MEMORY MANAGEMENT

MEMORY MANAGEMENT IS NOT UTILIZED IN THIS PROGRAM.

## 4.7 ERROR LOGGING

AT THE END OF EACH PASS ON ALL UNITS, THE PROGRAM PRINTS OUT THE CUMULATIVE TOTAL NUMBER OF ERRORS SINCE THE LAST START OR RESTART COMMAND.

## 5.0 PROGRAM LOAD MEDIA

THIS PROGRAM CAN BE LOADED FROM PAPER TAPE USING THE ABSOLUTE LOADER OR FROM ACT, SLIDE, OR APT SYSTEMS, OR FROM ANY MEDIA SUPPORTED BY XXDP+. WHEN USING THE PAPER TAPE ABSOLUTE LOADER, THE PROGRAM SHOULD BE LOADED FIRST, FOLLOWED BY THE DIAGNOSTIC SUPERVISOR. WHEN USING XXDP+, THE



CVDMAA.P11 12-DEC-80 15:59

PROGRAM DOCUMENT

215  
216

DIAGNOSTIC SUPERVISOR SHOULD BE LOADED FIRST, FOLLOWED BY  
THE DIAGNOSTIC PROGRAM.



CVDMAA.P11

12-DEC-80 15:59

## PROGRAM DOCUMENT

217  
218  
219  
220  
221  
222  
223  
224  
225  
226  
227  
228  
229  
230  
231  
232  
233  
234  
235  
236  
237  
238  
239  
240  
241  
242  
243  
244  
245  
246  
247  
248  
249  
250  
251  
252  
253  
254  
255  
256  
257  
258  
259  
260  
261  
262  
263  
264  
265  
266  
267  
268  
269  
270  
271  
272

## 6.0 OPERATING INSTRUCTIONS

## 6.1 LOADING AND STARTING PROCEDURES

## 6.1.1 LOADING PROCEDURES

THIS PROGRAM MAY BE LOADED FROM PAPER TAPE USING THE ABSOLUTE LOADER. IT MAY ALSO BE LOADED FROM ANY XXDP+ LOAD MEDIA. WHEN LOADED UNDER XXDP+, THE DIAGNOSTIC SUPERVISOR WILL BE LOADED AUTOMATICALLY.

## 6.1.2 STARTING PROCEDURES

THE PROGRAM STARTS AT LOCATION 20C. USE STANDARD DEC PROCEDURES TO START THE PROGRAM.

## 6.1.3 STEPS FOR QUICK AND SIMPLE EXECUTION

THE DIAGNOSTIC CAN BE EXECUTED STANDALONE UNDER XXDP+, WITHOUT READING THE REMAINDER OF THIS DOCUMENT, AS FOLLOWS:

- A) LOAD AND START DIAGNOSTIC USING RUN COMMAND
- B) RECEIVE DIAGNOSTIC SUPERVISOR IDENTIFICATION AND PROMPT (DRS-C>)
- C) ENTER STA<CR>
- D) ANSWER HARDWARE AND SOFTWARE QUESTIONS
- E) GET END OF PASS MESSAGES OR ERROR MESSAGES
- F) TO END EXECUTION, ENTER CONTROL/C

## 6.2 INITIAL DIALOGUE

AFTER THE PROGRAM AND THE SUPERVISOR ARE LOADED AND THE PROGRAM IS STARTED, THE FOLLOWING IDENTIFICATION IS TYPED :

DRS LOADED  
DIAG. RUN-TIME SERVICES  
CVDMA-A-0  
DMV-11 U-CONTRL LOGIC DIAG - PART 1 OF 2  
UNIT IS M8053 OR M8064  
DR>

THE OPERATOR THEN PROCEEDS BY TYPING ONE OR MORE OF THE COMMANDS DESCRIBED IN THE FOLLOWING SECTION 6.3. (FOR MORE DETAILED INFORMATION, REFER TO THE DIAGNOSTIC SUPERVISOR FUNCTIONAL SPECIFICATION).

## 6.3 PROGRAM OPTIONS



CVDMAA.P11

12-DEC-80 15:59

PROGRAM DOCUMENT

6.3.1 START COMMAND

```

*****
STA(RT)/TESTS:<TEST-LIST>/PASS:<PASS-CNT>/FLAGS:
<FLAG-LIST>/EOP:<INCR>
*****

```

6.3.1.1 TESTS SWITCH (/TESTS:<TEST-LIST>)

<TEST-LIST> IS A SEQUENCE OF DECIMAL NUMBERS (1:2 ETC.) OR RANGES OF DECIMAL NUMBERS (1-5:8-10 ETC.) THAT SPECIFY THE TESTS TO BE EXECUTED. THE NUMBERS ARE SEPARATED BY COLONS. THE NUMBERS RANGE FROM 1 TO THE LARGEST TEST NUMBER IN THE DIAGNOSTIC. THEY MAY BE SPECIFIED IN ANY ORDER. TESTS WILL BE EXECUTED IN NUMERICAL ORDER REGARDLESS OF THE ORDER OF SPECIFICATION. THE DEFAULT IS TO EXECUTE ALL TESTS. ON THIS AND ALL SWITCHES, THE ANGLE BRACKETS <> ARE PUNCTUATION USED IN THE DEFINITION ONLY, AND ARE NOT TO BE TYPED BY THE OPERATOR. SEE EXAMPLE AT END OF 6.3.1.5.

6.3.1.2 PASS SWITCH (/PASS:<PASS-CNT>)

<PASS-CNT> IS A DECIMAL NUMBER INDICATING THE DESIRED NUMBER OF PASSES. A PASS IS DEFINED AS THE EXECUTION OF THE FULL DIAGNOSTIC (ALL SELECTED TESTS) AGAINST ALL UNITS SUBMITTED. THE DEFAULT IS NON-ENDING EXECUTION. IN THIS CASE EXIT FROM THE PROGRAM IS ACCOMPLISHED EITHER BY TYPING A CONTROL/C OR BY OCCURANCE OF AN ERROR WITH THE HALT ON ERROR FLAG BEING SET. THE EXIT IS A RETURN TO COMMAND MODE. SEE EXAMPLE AT END OF 6.3.1.5.

6.3.1.3 FLAGS SWITCH (/FLAGS:<FLAG-LIST>)

<FLAG-LIST> IS A SEQUENCE OF ELEMENTS OF THE FORM <FLAG>, <FLAG=1>, OR <FLAG=0>, SEPARATED BY COLONS, WHERE <FLAG> HAS ONE OF THE FOLLOWING VALUES:

- HOE HALT ON ERROR, CAUSING COMMAND MODE TO BE ENTERED WHEN AN ERROR IS ENCOUNTERED
- LOE LOOP ON ERROR, CAUSING THE DIAGNOSTIC TO LOOP CONTINUOUSLY WITHIN THE SMALLEST DEFINED BLOCK OF CODING (SEGMENT, SUBTEST, OR TEST) CONTAINING THE ERROR
- IER INHIBIT ERROR REPORTING
- IBE INHIBIT BASIC ERROR REPORTS
- IXE INHIBIT EXTENDED ERROR REPORTS
- PRI DIRECT ALL MESSAGES TO A LINE PRINTER
- PNT PRINT NUMBER OF TEST BEING EXECUTED
- BOE BELL ON ERROR
- UAM RUN IN UNATTENDED MODE, BYPASSING MANUAL INTERVENTION TESTS
- ISR INHIBIT STATISTICAL REPORTS
- IDU INHIBIT DROPPING OF UNITS BY DIAGNOSTIC

273  
274  
275  
276  
277  
278  
279  
280  
281  
282  
283  
284  
285  
286  
287  
288  
289  
290  
291  
292  
293  
294  
295  
296  
297  
298  
299  
300  
301  
302  
303  
304  
305  
306  
307  
308  
309  
310  
311  
312  
313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323  
324  
325  
326  
327  
328



## PROGRAM DOCUMENT

## LOT LOOP ON TEST

THE FLAGS NAMED OR EQUATED TO 1 ARE SET, THOSE EQUATED TO 0 ARE CLEARED. A FLAG NOT SPECIFIED IS CLEARED. IF THE FLAGS SWITCH IS NOT GIVEN ALL FLAGS ARE CLEARED. SEE EXAMPLE AT END OF 6.3.1.5.

## 6.3.1.4 END OF PASS SWITCH (/EOP:&lt;INCR&gt;)

<INCR> IS A DECIMAL NUMBER INDICATING HOW OFTEN (IN TERMS OF PASSES) IT IS DESIRED THAT THE END OF PASS MESSAGE BE PRINTED. THE DEFAULT IS AT THE END OF EVERY PASS. SEE EXAMPLE AT END OF 6.3.1.5.

## 6.3.1.5 EFFECT OF START COMMAND

THE EFFECT OF THE START COMMAND IS TO INITIATE THE HARDWARE PARAMETER DIALOGUE, THE SOFTWARE PARAMETER DIALOGUE, AND THEN THE DIAGNOSTIC TESTS THEMSELVES.

THE HARDWARE PARAMETER DIALOGUE COMMENCES WITH THE QUESTION '# UNITS?' TO WHICH THE OPERATOR REPLIES WITH A DECIMAL NUMBER N FROM 1 TO 16. THE TERM 'UNIT' REFERS TO THE DEVICE TO WHICH THIS SERIES OF DIAGNOSTICS IS DEDICATED. FOLLOWING THIS ARE THE QUESTIONS WHEREBY THE P-TABLES THEMSELVES WILL BE BUILT. EACH P-TABLE IS A CORE-RESIDENT TABLE CONTAINING ALL THE HARDWARE INFORMATION FOR ONE UNIT. THE OPERATOR MUST SUPPLY N (NUMBER OF UNITS) VALUES FOR EACH QUESTION. HE MAY DO THIS BY GIVING ONE ANSWER TO EACH QUESTION (IN WHICH CASE THE SERIES OF QUESTIONS WILL BE POSED N TIMES) OR BY GIVING N VALUES, SEPARATED BY COMMAS, TO EACH QUESTION (SERIES WILL BE POSED ONCE). EACH QUESTION IS FOLLOWED BY THE RESPONSE RADIX (D FOR DECIMAL, B FOR BINARY, O FOR OCTAL, L FOR YES/NO) IN PARENTHESES AND THE DEFAULT VALUE AFTER THE PARENTHESES.

FOLLOWING THE HARDWARE QUESTIONS ARE THE SOFTWARE QUESTIONS TO BUILD THE SOFTWARE TABLES, WHICH DEFINE THE MODE (QUICK VERIFY ETC.) THAT THE DIAGNOSTIC WILL EXECUTE IN.

WHEN THE QUESTION '# UNITS?' IS ANSWERED, MEMORY STORAGE IS ALLOCATED FOR THE P-TABLES, AND IF THERE IS NOT ENOUGH TO ACCOMMODATE THEM THE MESSAGE 'TOO MANY UNITS' IS ISSUED. IN THIS CASE THE DIAGNOSTIC MUST BE EXECUTED MORE THAN ONCE TO TEST ALL UNITS.

## EXAMPLE:

STA/TESTS:1:2-4:6:8-10/PASS:3/FLAGS:IER:HOE=1:UAM:LOE

THIS COMMAND WILL CAUSE THREE PASSES TO BE MADE, EACH PASS CONSISTING OF TESTS 1,2,3,4,6,8,9, AND 10 EXECUTED AGAINST ALL UNITS. THERE IS NO DIFFERENCE BETWEEN SAYING <FLAG> AND SAYING <FLAG=1>. THE NOTATION <FLAG=0> IS MEANINGFUL ONLY ON

329  
330  
331  
332  
333  
334  
335  
336  
337  
338  
339  
340  
341  
342  
343  
344  
345  
346  
347  
348  
349  
350  
351  
352  
353  
354  
355  
356  
357  
358  
359  
360  
361  
362  
363  
364  
365  
366  
367  
368  
369  
370  
371  
372  
373  
374  
375  
376  
377  
378  
379  
380  
381  
382  
383  
384



PROGRAM DOCUMENT

A COMMAND OTHER THAN START TO CLEAR A FLAG THAT WAS PREVIOUSLY SET. NOTE THAT ON ALL COMMANDS ONLY THE FIRST THREE LETTERS ARE SCANNED.

6.3.2 RESTART COMMAND

\*\*\*\*\*
RES(TART)/TESTS:<TEST-LIST>/PASS:<PASS-CNT>/FLAGS:
<FLAG-LIST>/UNITS:<UNIT-LIST>
\*\*\*\*\*

6.3.2.1 TESTS, PASS, AND FLAGS SWITCHES

<TEST-LIST>, <PASS-CNT>, AND <FLAG-LIST> ARE AS IN THE START COMMAND.

6.3.2.2 UNITS SWITCH (/UNITS:<UNIT-LIST>)

<UNIT-LIST> IS A SEQUENCE OF DECIMAL NUMBERS (0,1 ETC.) OR RANGES OF DECIMAL NUMBERS (0-5, 8-10 ETC.) THAT SPECIFY THE UNITS TO BE TESTED. THE NUMBERS ARE SEPARATED BY COLONS. THE NUMBERS MAY RANGE FROM 0 THRU N-1 (N IS THE NUMBER OF UNITS SPECIFIED IN THE PREVIOUS START COMMAND). THE NUMBER INDICATES THE POSITION OF THE P-TABLE AS THE DATA WAS ENTERED DURING THE HARDWARE DIALOGUE. THE UNITS WHICH ARE SELECTED MUST NOT HAVE BEEN DROPPED BY THE DROP COMMAND. SEE THE DISCUSSION OF ADD AND DROP COMMANDS BELOW. DEFAULT IS TO TEST ALL UNITS WHICH HAVE NOT BEEN DROPPED BY A DROP COMMAND.

6.3.2.3 EFFECT OF RESTART COMMAND

THE RESTART COMMAND DIFFERS FROM THE START COMMAND IN THAT THE P-TABLES FROM THE PREVIOUS START COMMAND (THERE MUST HAVE BEEN ONE) ARE USED, INSTEAD OF NEW ONES BEING BUILT. THE UNITS SWITCH GIVES THE ABILITY TO SELECT A SUBSET OF THESE. THE SOFTWARE DIALOGUE MAY OPTIONALLY BE REEXECUTED (OPERATOR WILL BE ASKED). THE COMMAND CAN BE USED AFTER COMMAND MODE HAS BEEN REENTERED IN ANY OF THE THREE NORMAL WAYS: A) THE REQUESTED NUMBER OF PASSES HAVE BEEN MADE B) AN ERROR WAS ENCOUNTERED WITH THE HALT ON ERROR FLAG SET C) A CONTROL/C WAS ENTERED BY THE OPERATOR.

6.3.3 CONTINUE COMMAND

\*\*\*\*\*
CON(TINUE)/PASS:<PASS-CNT>/FLAGS:<FLAG-LIST>
\*\*\*\*\*

6.3.3.1 PASS SWITCH (/PASS:<PASS-CNT>)

385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440



## PROGRAM DOCUMENT

441  
442  
443  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479  
480  
481  
482  
483  
484  
485  
486  
487  
488  
489  
490  
491  
492  
493  
494  
495  
496

<PASS-CNT> IS SAME AS IN START COMMAND, BUT THE DEFAULT IS THE UNSATISFIED PASS-CNT FROM THE PREVIOUS START OR RESTART. IF NONE REMAINS, THE DEFAULT IS NON-ENDING EXECUTION.

## 6.3.3.2 FLAG SWITCH (/FLAGS:&lt;FLAG-LIST&gt;)

<FLAG-LIST> IS SAME AS IN START COMMAND, BUT UNSPECIFIED FLAGS RETAIN THEIR CURRENT VALUE.

## 6.3.3.3 EFFECT OF CONTINUE COMMAND

CONTINUE MUST FOLLOW A START OR RESTART, AND COMMAND MODE MUST HAVE BEEN ENTERED DUE TO A HALT ON ERROR OR A CONTROL/C. THE EFFECT OF THE COMMAND IS TO GO TO THE BEGINNING OF THE TEST THAT WAS BEING EXECUTED WHEN THE HALT OR CONTROL/C TOOK PLACE. SOFTWARE DIALOGUE MAY OPTIONALLY BE REEXECUTED. HARDWARE PARAMETERS MAY NOT BE CHANGED.

## 6.3.4 PROCEED COMMAND

```
*****  
PRO(CEED)/FLAGS:<FLAG-LIST>  
*****
```

## 6.3.4.1 FLAGS SWITCH (/FLAGS:&lt;FLAG-LIST&gt;)

<FLAG-LIST> IS AS IN THE START COMMAND, BUT UNSPECIFIED FLAGS RETAIN THEIR CURRENT VALUE.

## 6.3.4.2 EFFECT OF PROCEED COMMAND

PROCEED MUST FOLLOW A START, RESTART, OR CONTINUE. COMMAND MODE MUST HAVE BEEN ENTERED VIA A HALT ON ERROR. THE EFFECT OF THE COMMAND IS TO BEGIN EXECUTION AT THE LOCATION FOLLOWING THE ERROR CALL. NEITHER HARDWARE NOR SOFTWARE PARAMETERS MAY BE ALTERED.

## 6.3.5 ADD COMMAND

```
*****  
ADD/UNITS:<UNIT-LIST>  
*****
```

## 6.3.5.1 UNITS SWITCH (/UNITS:&lt;UNIT-LIST&gt;)

<UNIT-LIST> IS AS IN THE RESTART COMMAND.



## PROGRAM DOCUMENT

497  
498  
499  
500  
501  
502  
503  
504  
505  
506  
507  
508  
509  
510  
511  
512  
513  
514  
515  
516  
517  
518  
519  
520  
521  
522  
523  
524  
525  
526  
527  
528  
529  
530  
531  
532  
533  
534  
535  
536  
537  
538  
539  
540  
541  
542  
543  
544  
545  
546  
547  
548  
549  
550  
551  
552

## 6.3.5.2 EFFECT OF ADD COMMAND

THE UNITS SPECIFIED ARE ADDED TO THE TEST SEQUENCE. EACH UNIT MUST HAVE A P-TABLE IN MEMORY DUE TO AN EARLIER HARDWARE DIALOGUE. THIS COMMAND MUST BE FOLLOWED BY A RESTART OR CONTINUE. THE UNITS SWITCH MUST BE SPECIFIED. THE ADD COMMAND IS MEANINGFUL ONLY FOR UNITS THAT WERE PREVIOUSLY DROPPED.

## 6.3.6 DROP COMMAND

```
*****  
DRO(P)/UNITS:<UNIT-LIST>  
*****
```

## 6.3.6.1 UNITS SWITCH (/UNITS:&lt;UNIT-LIST&gt;)

<UNIT-LIST> IS AS IN THE RESTART COMMAND.

## 6.3.6.2 EFFECT OF DROP COMMAND

THE UNITS SPECIFIED WILL BE DROPPED FROM TESTING. THE UNITS WILL BE RESELECTED ONLY BY THE EXECUTION OF AN ADD OR START COMMAND. THE UNITS SWITCH MUST BE ENTERED. THIS COMMAND MUST BE FOLLOWED BY A RESTART OR A CONTINUE COMMAND.

## 6.3.7 PRINT COMMAND

```
*****  
PRI(NT)  
*****
```

## 6.3.7.1 EFFECT OF PRINT COMMAND

THE TOTAL NUMBER OF ERRORS FOR EACH UNIT SINCE THE LAST START OR RESTART COMMAND ARE PRINTED. THE ISR (INHIBIT STATISTICAL REPORTING) FLAG IS CLEARED.

## 6.3.8 DISPLAY COMMAND

```
*****  
DIS(PLAY)/UNITS:<UNIT-LIST>  
*****
```

## 6.3.8.1 UNITS SWITCH (/UNITS:&lt;UNIT-LIST&gt;)

<UNIT-LIST> IS AS IN THE RESTART COMMAND.



## PROGRAM DOCUMENT

553  
554  
555  
556  
557  
558  
559  
560  
561  
562  
563  
564  
565  
566  
567  
568  
569  
570  
571  
572  
573  
574  
575  
576  
577  
578  
579  
580  
581  
582  
583  
584  
585  
586  
587  
588  
589  
590  
591  
592  
593  
594  
595  
596  
597  
598  
599  
600  
601  
602  
603  
604  
605  
606  
607  
608

## 6.3.8.2 EFFECT OF DISPLAY COMMAND

THE HARDWARE P-TABLES FOR ALL UNITS UNDER TEST ARE PRINTED OUT IN THE FORMAT IN WHICH THEY WERE ENTERED. ANY UNITS THAT WERE DROPPED BY THE OPERATOR 'DROP' COMMAND ARE SO DESIGNATED.

## 6.3.9 FLAGS COMMAND

\*\*\*\*\*  
FLA(GS)  
\*\*\*\*\*

## 6.3.9.1 EFFECT OF FLAGS COMMAND

THE CURRENT SETTINGS OF ALL FLAGS ARE PRINTED.

## 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- HARD CORE QUESTIONS (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 SUPPRESSED FOR THE REMAINDER OF THE DIAGNOSTIC OR UNTIL ANOTHER O IS TYPED, WHICH RESTORES NORMAL TELETYPE OUTPUT.

## 6.3.12 HARDWARE PARAMETERS

THE FOLLOWING 3 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.



PROGRAM DOCUMENT

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  
659  
660  
661  
662  
663  
664

1. DEVICE CSR ADDRESS : (0) 160020?

THIS IS THE ADDRESS AT WHICH THE CSR REGISTERS (SELO) RESIDE ON THE LSI-BUS. THE ALLOWABLE RANGE IS 160020-177760 (OCTAL), AND THE DEFAULT VALUE IS 160020.

2. DEVICE VECTOR ADDRESS : (0) 300 ?

THIS IS THE ADDRESS OF THE INPUT INTERRUPT VECTOR FOR THIS DEVICE. THE ALLOWABLE RANGE IS 000-674 (OCTAL), AND THE DEFAULT VALUE IS 300.

3. DEVICE PRIORITY LEVEL : (0) 4 ?

THIS IS THE CPU PRIORITY AT WHICH THE INTERRUPT HANDLERS OF THIS DEVICE WILL BE EXECUTED. THE ALLOWABLE RANGE IS 0-7, AND THE DEFAULT VALUE IS 4.

6.3.13 SOFTWARE PARAMETERS

NO SOFTWARE PARAMETER QUESTIONS ARE ASKED BY PART 1 OF THE 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.



CVDMAA.P11

12-DEC-80 15:59

## PROGRAM DOCUMENT

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  
716  
717

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).

NOW LET US SEE HOW WE COULD USE THESE CAPABILITIES TO CONSTRUCT A SET OF P-TABLES. ASSUME THAT WE HAVE 16 UNITS, AND THAT THERE ARE THREE HARDWARE PARAMETERS FOR EACH (THREE SLOTS IN THE P-TABLE, THREE HARDWARE QUESTIONS IN THE DIALOGUE). LET THE DESIRED VALUE FOR THE FIRST PARAMETER BE THE NUMBER 75 FOR ALL 16 TABLES. LET THE DESIRED VALUE FOR THE SECOND PARAMETER BE EQUAL TO THE UNIT NUMBER (0,1,2,...,15) EXCEPT FOR UNIT 12, WHICH SHOULD RECEIVE THE VALUE 11. LET THE DESIRED VALUE FOR THE THIRD PARAMETER BE THE NUMBER 76 FOR THE FIRST 7 UNITS AND THE NUMBER 77 FOR THE LAST 9 UNITS.

THE FOLLOWING DIALOGUE WOULD ACCOMPLISH THIS GOAL:

```
# UNITS (D) ? 16
UNIT 0
<QUESTION 1> ? 75
<QUESTION 2> ? 0-6
<QUESTION 3> ? 76

UNIT 7
<QUESTION 1> ?
<QUESTION 2> ? 7-11,,13-15
<QUESTION 3> ? 77
```

THE FIRST TIME THE SERIES IS ASKED, SLOT ONE RECEIVES A 75 IN ALL 16 TABLES. SLOT TWO RECEIVES THE VALUES 0,1,2,...,6 IN TABLES 0 THRU 6 AND A CONSTANT 6 IN TABLES 7 THRU 15. SLOT THREE RECEIVES A CONSTANT 76 IN ALL 16 TABLES.

THE SECOND TIME THRU THE SERIES, TABLES 7 THRU THE END ARE GOING TO BE AFFECTED (NOTE THAT THIS PIECE OF INFORMATION IS PRINTED OUT FOR THE OPERATOR IN THE FORM 'UNIT XX' AT THE BEGINNING OF EACH SERIES). QUESTION 1 IS RESPONDED TO BY A <CR>, SO SLOT ONE STAYS AT CONSTANT 75 IN TABLES 7 THRU 15, SINCE NO NEW EXPLICIT VALUES ARE TYPED IN. SLOT TWO GETS THE VALUES 7,8,9,10,11 IN TABLES 7 THRU 11, AND GETS AN 11 IN SLOT 12, AND GETS THE VALUES 13,14,15 IN TABLES 13 THRU 15. SLOT THREE GETS THE VALUE 77 IN TABLES 7 THRU 15.

THE DIALOGUE IS TERMINATED WHEN THE SOFTWARE RECOGNIZES THAT 16 EXPLICIT VALUES HAVE BEEN GIVEN FOR AT LEAST ONE QUESTION (NAMELY QUESTION 2).



718  
719  
720  
721  
722  
723  
724  
725  
726  
727  
728  
729  
730  
731  
732  
733  
734  
735  
736  
737  
738  
739  
740  
741  
742  
743  
744  
745  
746  
747  
748  
749  
750  
751  
752  
753  
754  
755  
756  
757  
758  
759  
760  
761  
762  
763  
764  
765  
766  
767  
768  
769  
770  
771  
772  
773

8.0 TEST DESCRIPTIONS

```

*****
* TEST 1 <DMV-11 AVAILABILITY>
*
* EACH NORMALLY USED CSR IS ACCESSED WITH A 'TST' OR 'TSTB' INSTRUCTION AND IF
* A BUS TIMEOUT OCCURS (INTERRUPT @ VECTOR ADDR 4) A FLAG WILL BE SET SHOWING
* WHICH CSR ADDR AND INSTRUCTION FAILED. 'T1.HSW' REFLECTS 'TST' INSTRUCTIONS
* AND 'T1.HSB' REFLECTS 'TSTB' INSTRUCTIONS.
*
* EXAMPLES:
*
* IF 'TSTB @SEL1' FAILS, BIT # 1 OF 'T1.HSB' WILL BE SET.
* IF 'TST @SEL4' FAILS, BIT # 4 OF 'T1.HSW' WILL BE SET
* (NOTE: ONLY EVEN BITS IN 'T1.HSW' CAN BE SET).
*
* THE FLAG WORDS ARE OUTPUT IN BINARY AS 'EXTENDED ERROR INFORMATION'.
*****

```

```

*****
* TEST 2 <MASTER CLEAR, RUN MICRODIAGNOSTICS>
*
* A MASTER CLEAR IS ISSUED TO THE DMV-11, AND THE PROGRAM ALLOWS SUFFICIENT
* TIME FOR THE MICRODIAGNOSTICS TO BE PERFORMED. THESE MICRODIAGNOSTICS RESIDE
* IN 6502 PROGRAM MEMORY, AND THOROUGHLY VERIFY THE OPERATION OF THE 6502
* MICROPROCESSOR CHIP. THEN, THEY CHECK OUT THE DATA RAM, THE 6502'S ACCESS TO
* THE CSR'S, AND PERFORM A SIMPLE MESSAGE TEST USING THE 6522 CHIP AND THE
* USYRT, WITH INTERNAL LOOPBACK.
*
* NEXT, THE LSI-11 PROGRAM READS THE THE CSR'S (SELO-SEL6) AND CHECKS THEM FOR
* THEIR EXPECTED INITIALIZED STATES. IF AN ERROR HAS OCCURRED IN THE MICRO-
* DIAGNOSTICS THE NUMBER OF THE FAILING TEST WILL BE FOUND IN SEL4, AND RUN
* (BIT 7) WILL NOT BE SET IN BSEL1.
*****

```

```

*****
* TEST 3 <CSR ADDRESSING>
*
* FIRST, HALT THE 6502 UP BY CLEARING ALL CSRS. THEN, WRITE A DIFFERENT WORD
* OF DATA PATTERN A INTO EACH OF BSEL0-17, AND AFTER EACH WRITE, READ AND
* COMPARE ALL REGS TO EXPECTED VALUES.
*
* DATA PATTERN A = 001, 002, 004, 010, 020, 040, 100, 200, 052, 300, 140,
*                   060, 030, 014, 006, 003
*****

```

```

*****
* TEST 4 <CSR REGISTERS DATA READ/WRITE>
*
* WRITE, READ, AND COMPARE EACH BYTE OF DATA PATTERN B INTO REGISTER BSEL0.

```



CVDMAA.P11 12-DEC-80 15:59

PROGRAM DOCUMENT

774  
775  
776  
777  
778  
779  
780  
781  
782  
783  
784  
785  
786  
787  
788  
789  
790  
791  
792  
793  
794  
795  
796  
797  
798  
799  
800  
801  
802  
803  
804  
805  
806  
807  
808  
809  
810  
811  
812  
813  
814  
815  
816  
817  
818  
819  
820  
821  
822  
823  
824  
825  
826  
827  
828  
829

:\* THEN, REPEAT THIS USING EACH OF THE REMAINING CSR'S, BSEL1-BSEL17. WHEN BSEL1  
:\* IS BEING TESTED, THE PROGRAM ALWAYS SETS BIT 7 IN THE DATA PATTERN SO THAT  
:\* RUN WILL NOT BE CLEARED, AND IT ALWAYS CLEARS BIT6 SO THAT MCLR WILL NOT BE  
:\* SET.

:\* DATA PATTERN B = 125, 252, 000, 377, 001, 002, 004, 010, 020, 040, 100,  
:\* 200, 376, 375, 373, 367, 357, 337, 277, 177, 000  
:\*\*\*\*\*

:\*\*\*\*\*  
:\* TEST 5 <BASIC MASTER CLEAR>

:\* PERFORM AN INITIAL MASTER CLEAR. WRITE 377 INTO BSEL0 AND READ AND CHECK IT.  
:\* THEN, ISSUE A MASTER CLEAR AND READ AND CHECK BSEL0 FOR 000.  
:\*\*\*\*\*

:\*\*\*\*\*  
:\* TEST 6 <BUS RESET>

:\* PERFORM AN INITIAL MASTER CLEAR. WRITE 377 INTO BSEL0 AND READ AND CHECK  
:\* IT. THEN, ISSUE A RESET INSTRUCTION, STALL FOR COMPLETION, AND READ AND  
:\* CHECK BSEL0 FOR 000.  
:\*\*\*\*\*

:\*\*\*\*\*  
:\* TEST 7 <CSR, MAINTENANCE MICROCODE INTERACTION>

:\* THIS TEST INVOKES THE MAINTENANCE REQUEST MECHANISM THROUGH WHICH THE LSI-11  
:\* AND 6502 CAN COMMUNICATE. FIRST, A MASTER CLEAR IS DONE WITH ONLY BIT 0  
:\* (MREQ) SET IN BSEL1. THE PROGRAM THEN CHECKS FOR THE SETTING OF BSEL2 BIT 7  
:\* (MRDY) BY THE MAINTENANCE MICROCODE WITHIN ABOUT 50 MICRO-SEC., AND IF MRDY  
:\* DOES NOT GET SET, AN ERROR IS REPORTED.

:\* NEXT, THE PROGRAM LOADS SEL4 WITH 000010 AND BSEL6 WITH 125. THEN, ALL CSR'S  
:\* ARE READ AND CHECKED FOR EXPECTED CONTENTS.

:\* BSEL2 IS THEN LOADED WITH A WRITE COMMAND, WHICH SHOULD CAUSE THE MICROCODE  
:\* TO TRANSFER THE 125 INTO BSEL0. ALL CSR'S ARE THEN READ AND CHECKED FOR  
:\* EXPECTED CONTENTS.

:\* THEN, THE PROGRAM LOADS 252 INTO BSEL0 AND READS AND CHECKS ALL CSR'S. BSEL2  
:\* IS THEN LOADED WITH A READ COMMAND, WHICH SHOULD CAUSE THE MICROCODE TO  
:\* TRANSFER THE 252 INTO BSEL6. ALL CSR'S ARE READ AND CHECKED.  
:\*\*\*\*\*

:\*\*\*\*\*  
:\* TEST 8 <RUN FLIP-FLOP>

:\* THE PROGRAM PUTS THE MICROCODE INTO THE MAINTENANCE LOOP. A 125 CHARACTER  
:\* IS LOADED INTO BSEL6 AND A REQUEST IS MADE TO WRITE THE CONTENTS OF BSEL6  
:\* INTO BSEL0. THE PROGRAM THEN READS AND CHECKS BSEL0 TO CONTAIN 125.  
:\* NEXT, THE RUN FLIP-FLOP IS CLEARED BY LOADING A 0 INTO RUN (BSEL1 BIT 7).



CVDMAA.P11

12-DEC-80 15:59

PROGRAM DOCUMENT

830  
831  
832  
833  
834  
835  
836  
837  
838  
839  
840  
841  
842  
843  
844  
845  
846  
847  
848  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
860  
861  
862  
863  
864  
865  
866  
867  
868  
869  
870  
871  
872  
873  
874  
875  
876  
877  
878  
879  
880  
881  
882  
883  
884  
885

\*\*\*\*\*  
: \* BSEL0 IS THEN CLEARED AND THE REQUEST IS MADE AGAIN TO WRITE THE CONTENTS  
: \* OF BSEL6 INTO BSEL0. THE PROGRAM STALLS FOR 50 MICRO-SEC. AND CHECKS FOR  
: \* MRDY (BSEL2 BIT 7) NOT SET, AND BSEL0 STILL CLEARED.  
: \* THEN, THE PROGRAM SETS THE RUN FLIP-FLOP AGAIN BY LOADING A 1 INTO RUN,  
: \* AND CHECKS FOR MRDY SET WITHIN 50 MICRO-SEC. AND BSEL0 = 125.  
\*\*\*\*\*

\*\*\*\*\*  
: \* TEST 9 <LOW RAM (00-0F) SCRATCHPAD>  
: \*  
: \* THIS TEST FIRST PERFORMS AN ADDRESSING TEST OF RAM LOCATIONS (00-0F), BY  
: \* WRITING THE ADRS INTO EACH LOCATION AND AFTER EACH WRITE, ALL THE LOCATIONS  
: \* ARE READ AND CHECKED FOR EXPECTED CONTENTS.  
: \*  
: \* THEN, THE TEST PERFORMS READ/WRITE DATA TESTING OF RAM LOCATIONS 00-0F,  
: \* BY WRITING, READING, AND COMPARING ALL BYTES OF DATA PATTERN B IN EACH  
: \* LOCATION.  
: \* DATA PATTERN B = 125, 252, 000, 377, 001, 002, 004, 010, 020, 040, 100,  
: \* 200, 376, 375, 373, 367, 357, 337, 277, 177, 000  
\*\*\*\*\*

\*\*\*\*\*  
: \* TEST 10 <DATA RAM MOVING INVERSIONS (LOC'S 0018-01FF HEX)>  
: \*  
: \* GENERAL DESCRIPTION:  
: \* FIRST, THE 2K BYTE LOCATIONS IN RAM ARE LOADED WITH 0'S (SEE NOTE BELOW).  
: \* THEN, THE FIRST LOCATION IS READ AND CHECKED, A SINGLE 1 IS WRITTEN INTO  
: \* THE LOW BIT POSITION, AND THIS IS READ AND CHECKED. THIS IS DONE FOR ALL  
: \* BYTES IN THE RAM, BY INCREMENTING THE ADDRESS TO POINT TO THE NEXT RAM  
: \* LOCATION.  
: \* THEN, THE NEXT BIT POSITION IS CHOSEN TO INSERT A 1, AND ALL LOCATIONS  
: \* ARE READ, WRITTEN, AND READ AS BEFORE. THIS IS CONTINUED FOR ALL BIT  
: \* POSITIONS UNTIL THE ENTIRE RAM IS WRITTEN TO ALL 1'S. THE ABOVE OPERATIONS  
: \* ARE PERFORMED A SECOND TIME, WITH 0'S INSERTED INTO THE RAM INSTEAD OF 1'S.  
: \* THIS RESULTS IN THE ENTIRE RAM BEING WRITTEN TO ALL 0'S.  
: \* THIS TEST CONSTITUTES A THOROUGH TEST OF THE RAM. IT IS CAPABLE OF  
: \* DETECTING THE FOLLOWING FAULTS : STUCK ADDRESS BITS, UNI- AND BI-DIRECT-  
: \* IONAL COUPLING BETWEEN ADDRESS BITS, STUCK MEMORY BITS, AND UNI- AND  
: \* BI-DIRECTIONAL COUPLING BETWEEN MEMORY BITS IN BOTH ROWS AND COLUMNS OF THE  
: \* MEMORY MATRIX.  
: \*  
: \* NOTE:  
: \* THIS TEST DOES NOT CHECK LOCATIONS 0010-001F, SO THAT THE PRIMARY CSR'S  
: \* ARE NOT WRITTEN. IT DOES TEST LOCATIONS 0000-000F (SCRATCHPAD RAM) AND  
: \* LOCATIONS 0020-002F (SECONDARY CSR'S), AS WELL AS 0030-0800 (BASIC RAM).  
: \*  
: \* THE 'TMP#' REGISTERS ARE USED HERE TO CONTAIN THE VARIOUS CONSTANTS &  
: \* VARIABLES USED THROUGHOUT THIS TEST. A LIST OF THEIR ASSIGNMENTS SEEMS  
: \* USEFUL SO HERE IT IS:  
: \*  
: \* TMP0 POINTS TO THE FIRST LOCATION AFTER THE SELECT REGISTERS.  
: \*  
: \* TMP1 ----  
: \*



CVDMAA.P11 12-DEC-80 15:59

PROGRAM DOCUMENT

886  
887  
888  
889  
890  
891  
892  
893  
894  
895  
896  
897  
898  
899  
900  
901  
902  
903  
904  
905  
906  
907  
908  
909  
910  
911  
912  
913  
914  
915  
916  
917  
918  
919  
920  
921  
922  
923  
924  
925  
926  
927  
928  
929  
930  
931  
932  
933  
934  
935  
936  
937  
938  
939  
940  
941

```

:*      TMP2  TEST PATTERN ID CODE -- UNUSED BY THIS TEST.
:*
:*      TMP3  TEST DATA PATTERN INDEX -- UNUSED BY THIS TEST.
:*
:*      TMP4  TEST DATA PATTERN.  THE HIGH BYTE IS THE PATTERN BEING WRITTEN
:*           ON ANY GIVEN PASS AND THE LOW BYTE IS THE PATTERN THAT WAS
:*           WRITTEN BY THE PREVIOUS PASS THROUGH THE RAM.
:*
:*      TMP5  DATA READ FROM THE RAM.  ONLY THE LOW BYTE IS USED.
:*
:*      TMP6  ----
:*      TMP7  ----
:*      TMP8  ----
:*      TMP9  ----
:*
:*      TMPA  RAM ADDRESS BEING TESTED.
:*
:*      TMPB  BIT POINTER.  NUMBER OF THE BIT WITHIN THE DATA FIELD WHICH IS
:*           BEING SWITCHED ON EACH WRITE WITHIN THE CURRENT PASS.
:*
:*      TMPC  DATA FLAG.  BIT 0 OF THIS WORD IS THE VALUE TO WHICH THE BIT
:*           IDENTIFIED IN TMPB IS BEING SET ON EACH WRITE IN THE CURRENT
:*           PASS.
:*
:*      TMPD  DIRECTION SWITCH.  0 = FORWARD    NON-ZERO = BACKWARD
:*
:*      TMPE  LAST VALID ADDRESS TO BE TESTED.  (I.E. THE END OF RAM)
:*
:*      TMPF  ERROR FLAGS.  BIT 1 SET = THE LAST DETECTED ERROR WAS THE READ
:*           OF THE PREVIOUS DATA BEFORE WRITING THE NEW DATA.  IF BIT2 IS
:*           SET, THE READ AFTER WRITE FAILED.  IF EITHER IS SET WHEN AN
:*           ERROR IS DETECTED, THE SUPERVISOR IS NOT CALL'D AND THEREFOR
:*           IT'S ERROR COUNTER WILL NOT REFLECT THE ERROR -- INSTEAD, THE
:*           DATA LINE IS PRINTED.  (UNLESS THE ERROR HANDLER'S DATA LINE
:*           PRINT COUNT HAS EXCEEDED ITS LIMIT -- IN WHICH CASE ITS
:*           INVOCATION IS IGNORED.)
:*****

```

```

:*****
:*      TEST 11 <VIA REGISTER ADDRESSING>
:*
:*      VIA == '6522 VERSATILE INTERFACE ADAPTER'
:*
:*      A MASTER CLEAR IS PERFORMED, NEXT, TIMER 1 LATCHES
:*      ARE CLEARED BY WRITING 000 INTO VIA REGS 6 & 7
:*      THEN, 377 IS LOADED INTO DATA DIRECTION REGISTERS A, B (DDRA, DDRB) TO
:*      SET THE PORT PINS FOR OUTPUT MODE.
:*      THEN, A DIFFERENT BYTE OF DATA PATTERN C IS WRITTEN INTO EACH VIA
:*      LOCATION, (EXCEPT THE TIMER REGS 4,5,10,11 OCT) AND AFTER EACH IS WRITTEN,
:*      ALL VIA REGS (EXCEPT 4,5,10,11) ARE READ AND COMPARED TO EXPECTED
:*      CONTENTS.  NOTE THAT SOME VIA REGS ARE ALTERED BY THE LOADING OF OTHERS,
:*      AND THE PROGRAM TAKES THIS INTO ACCOUNT, IN THE SETTING OF EXPECTED REG
:*      VALUES.  THE DATA PATTERN IS CHOSEN TO AVOID ACTIVATING THE VIA CHIP (SUCH
:*      AS GENERATING OUTPUTS ON CA1, CA2, CB1, CB2, OR CAUSING 6502
:*      INTERRUPT REQUESTS).

```



## PROGRAM DOCUMENT

942  
943  
944  
945  
946  
947  
948  
949  
950  
951  
952  
953  
954  
955  
956  
957  
958  
959  
960  
961  
962  
963  
964  
965  
966  
967  
968  
969  
970  
971  
972  
973  
974  
975  
976  
977  
978  
979  
980  
981  
982  
983  
984  
985  
986  
987  
988  
989  
990  
991  
992  
993  
994  
995  
996  
997

```

: * DATA PATTERN C (WITH VIA REGS AND THEIR DATA SHOWN IN OCTAL) :
: * REGISTER = 00 01 02 03 06 07 12 13 14 15 16 17
: * DATA = 100, 101, 377, 377, 106, 107, 112, 040, 042, 000, 200, 117
: * NEXT, 000 IS LOADED INTO DDRA, AND DDRB IS READ AND COMPARED TO 377. THEN,
: * THE 377 IS LOADED BACK INTO DDRA, AND DDRB IS LOADED WITH 000 AND DDRA IS
: * READ AND COMPARED TO 377.
: *****
: *****
: * TEST 12 <VIA'S DDRB DATA READ/WRITE>
: *
: * DDRB == 'DATA DIRECTION REGISTER B'
: * FIRST, A MASTER CLEAR IS PERFORMED. THEN :
: * READ/WRITE BITS 0-7 OF VIA DATA DIRECTION REGISTER B ARE TESTED BY WRITING,
: * READING, AND COMPARING EACH BYTE OF DATA PATTERN B.
: * DATA PATTERN B = 125, 252, 000, 377, 001, 002, 004, 010, 020, 040, 100,
: *                   200, 376, 375, 373, 367, 357, 337, 277, 177, 000
: *****
: *****
: * TEST 13 <VIA'S DDRA DATA READ/WRITE>
: *
: * DDRA == 'DATA DIRECTION REGISTER A'
: * FIRST, A MASTER CLEAR IS PERFORMED. THEN :
: * READ/WRITE BITS 0-7 OF VIA DATA DIRECTION REGISTER A ARE TESTED BY WRITING,
: * READING, AND COMPARING EACH BYTE OF DATA PATTERN B.
: * DATA PATTERN B = 125, 252, 000, 377, 001, 002, 004, 010, 020, 040, 100,
: *                   200, 376, 375, 373, 367, 357, 337, 277, 177, 000
: *****
: *****
: * TEST 14 <VIA'S ORB DATA READ/WRITE>
: *
: * ORB == 'OUTPUT REGISTER PORT B'
: * FIRST, A MASTER CLEAR IS PERFORMED. NEXT, 377 IS LOADED INTO DATA
: * DIR. REG. B (DDRb) TO SET ALL B PORT PINS FOR OUTPUT MODE. THEN
: * READ/WRITE BITS 0-7 OF VIA OUTPUT REG. PORT B ARE TESTED BY WRITING,
: * READING, AND COMPARING EACH BYTE OF DATA PATTERN B.
: * DATA PATTERN B = 125, 252, 000, 377, 001, 002, 004, 010, 020, 040, 100,
: *                   200, 376, 375, 373, 367, 357, 337, 277, 177, 000
: *****
: *****
: * TEST 15 <VIA'S T1 DATA READ/WRITE>
: *
: * T1 == 'TIMER #1'
: * THIS TEST WRITES, READS, AND CHECKS THE T1 LATCHES AND COUNTER REGISTERS
: * WITH DATA PATTERNS IN EACH OF 3 SUBTESTS.
: *

```



CVDMAA.P11

12-DEC-80 15:59

PROGRAM DOCUMENT

998  
999  
1000  
1001  
1002  
1003  
1004  
1005  
1006  
1007  
1008  
1009  
1010  
1011  
1012  
1013  
1014  
1015  
1016  
1017  
1018  
1019  
1020  
1021  
1022  
1023  
1024  
1025  
1026  
1027  
1028  
1029  
1030  
1031  
1032  
1033  
1034  
1035  
1036  
1037  
1038  
1039  
1040  
1041  
1042  
1043  
1044  
1045  
1046  
1047  
1048  
1049  
1050  
1051  
1052  
1053

```

: *
: * FIRST SUBTEST: CHECKS FOR PROPER LOADING OF LATCHES
: *   IT ALSO CHECKS TO BE SURE THAT THE COUNTER APPEARS TO BE DOING
: *   SOMETHING TO THE COUNTERS. AS LONG AS THEY HAVE CHANGED FROM THE
: *   VALUE LOADED INTO THEM, WE WILL BE SATISFIED.
: *
: * A. A MASTER CLEAR IS PERFORMED.
: * B. ALL LATCHES ARE LOADED TO ZEROES (JUST IN CASE), ACR6 & ACR7 ARE SET
: *   TO ZERO (MODE 00), AND 'T1' INTERRUPT ENABLE FLAG IS CLEARED.
: *
: * C. T1L-L(ADR 04) IS LOADED WITH THE CURRENT BYTE OF DATA PATTERN B.
: * D. T1L-L(ADR 06) IS READ AND COMPARED TO THE BYTE JUST WRITTEN.
: * E. T1C-L(ADR 04) IS READ AND CHECKED TO BE DIFFERENT THAN THE TEST BYTE.
: *
: * F. T1L-L(ADR 06) IS LOADED WITH THE COMPLEMENT OF THE CURRENT DATA BYTE.
: * G. T1L-L(ADR 06) IS READ AND COMPARED TO THE BYTE JUST WRITTEN.
: *
: * H. T1L-L(ADR 06) IS RE-LOADED WITH 0 TO MAKE T1C-H DECREMENT FAST.
: *   T1L-H(ADR 05) IS LOADED WITH THE ORIGINAL TEST DATA PATTERN BYTE.
: * I. T1L-H(ADR 07) IS READ AND COMPARED TO THE BYTE LOADED INTO T1L-H.
: *
: * J. T1C-H(ADR 05) IS READ AND CHECKED TO BE DIFFERENT THAN THE TEST BYTE.
: *
: * K. T1L-H(ADR 07) IS LOADED WITH THE COMPLEMENT OF THE CURRENT DATA BYTE.
: * L. T1L-H(ADR 07) IS READ AND COMPARED TO THE BYTE JUST LOADED.
: *
: * M. STEPS C-L ARE REPEATED USING EACH BYTE OF DATA PATTERN B.
: *
: * SECOND SUBTEST: CHECKS FOR CROSS-TALK AND ADDRESSING ERRORS
: *   FROM T1L-L TO T1L-H
: *
: * A. T1L-H(ADR 07) IS LOADED WITH 000 TO CLEAR IT.
: * B. T1L-L(ADR 06) IS LOADED WITH A BYTE OF DATA PATTERN B.
: * C. T1L-L(ADR 06) IS READ AND COMPARED TO THE DATA JUST WRITTEN.
: * D. T1L-H(ADR 07) IS READ AND COMPARED TO 000.
: * E. STEPS B-D ARE REPEATED USING EACH BYTE OF DATA PATTERN B.
: *
: * THIRD SUBTEST: CHECKS FOR CROSS-TALK AND ADDRESSING ERRORS
: *   FROM T1L-H TO T1L-L
: *
: * A. T1L-L(ADR 04) IS LOADED WITH 000 TO CLEAR IT
: * B. T1L-H(ADR 07) IS LOADED WITH A BYTE OF DATA PATTERN B.
: * C. T1L-H(ADR 07) IS READ AND COMPARED TO THE DATA JUST WRITTEN.
: * D. T1L-L(ADR 06) IS READ AND COMPARED TO 000.
: * E. STEPS B-D ARE REPEATED USING EACH BYTE OF DATA PATTERN B.
: *
: *   DATA PATTERN B = 125, 252, 000, 377, 001, 002, 004, 010, 020, 040, 100,
: *                   200, 376, 375, 373, 367, 357, 337, 277, 177, 000
: * *****
: *
: * *****
: * TEST 16 <VIA'S SR DATA READ/WRITE>
: *
: *

```



PROGRAM DOCUMENT

1054  
1055  
1056  
1057  
1058  
1059  
1060  
1061  
1062  
1063  
1064  
1065  
1066  
1067  
1068  
1069  
1070  
1071  
1072  
1073  
1074  
1075  
1076  
1077  
1078  
1079  
1080  
1081  
1082  
1083  
1084  
1085  
1086  
1087  
1088  
1089  
1090  
1091  
1092  
1093  
1094  
1095  
1096  
1097  
1098  
1099  
1100  
1101  
1102  
1103  
1104  
1105  
1106  
1107  
1108  
1109

```

: *      SR == 'SHIFT REGISTER'
: *
: * FIRST, A MASTER CLEAR IS PERFORMED AND THE ACR IS SET TO 000. THEN :
: * READ/WRITE BITS 0-7 OF VIA SHIFT REGISTER ARE TESTED BY WRITING, READING,
: * AND COMPARING EACH BYTE OF DATA PATTERN B.
: * DATA PATTERN B = 125, 252, 000, 377, 001, 002, 004, 010, 020, 040, 100,
: *                   200, 376, 375, 373, 367, 357, 337, 277, 177, 000
: *****

: *
: * TEST 17 <VIA'S ACR DATA READ/WRITE>
: *
: * ACR == 'AUXILIARY CONTROL REGISTER'
: *
: * FIRST, A MASTER CLEAR IS PERFORMED. THEN :
: * READ/WRITE BITS 0-7 OF THE ACR ARE TESTED BY WRITING, READING,
: * AND COMPARING EACH BYTE OF DATA PATTERN B.
: * DATA PATTERN B = 125, 252, 000, 377, 001, 002, 004, 010, 020, 040, 100,
: *                   200, 376, 375, 373, 367, 357, 337, 277, 177, 000
: *****

: *
: * TEST 18 <VIA'S PCR DATA READ/WRITE>
: *
: * PCR == 'PERIPHERAL CONTROL REGISTER'
: *
: * FIRST, A MASTER CLEAR IS PERFORMED. THEN :
: * READ/WRITE BITS 0-7 OF THE PCR REGISTER ARE TESTED BY WRITING, READING,
: * AND COMPARING EACH BYTE OF A SUBSET OF DATA PATTERN B.
: * DATA PATTERN B (SUBSET) = 125, 252, 000, 377, 001, 002, 004, 010, 020,
: *                             040, 100, 200.
: *****

: *
: * TEST 19 <VIA'S IER DATA READ/WRITE>
: *
: * IER == 'INTERRUPT ENABLE REGISTER'
: *
: * BITS 0-6 OF THE IER CAN BE SET OR CLEARED ON A WRITE, UNDER CONTROL OF THE
: * SET/CLEAR CONTROL BIT 7. TO TEST THIS, EACH BYTE OF DATA PATTERN D IS
: * WRITTEN INTO IER, AND THE REGISTER IS READ AND COMPARED TO THE CORRESPOND-
: * ING BYTE OF DATA PATTERN E.
: *
: * DATA PATTERN D = 200, 201, 202, 204, 210, 220, 240, 300, 200, 000, 001,
: *                   002, 004, 010, 020, 040, 100, 000, 325, 125, 252, 052
: *
: * DATA PATTERN E = 000, 001, 003, 007, 017, 037, 077, 177, 177, 177, 176,
: *                   174, 170, 160, 140, 100, 000, 000, 125, 000, 052, 000
: *****
: *****

```

PROGRAM DOCUMENT

1110  
1111  
1112  
1113  
1114  
1115  
1116  
1117  
1118  
1119  
1120  
1121  
1122  
1123  
1124  
1125  
1126  
1127  
1128  
1129  
1130  
1131  
1132  
1133  
1134  
1135  
1136  
1137  
1138  
1139  
1140  
1141  
1142  
1143  
1144  
1145  
1146  
1147  
1148  
1149  
1150  
1151  
1152  
1153  
1154  
1155  
1156  
1157  
1158  
1159  
1160  
1161  
1162  
1163  
1164  
1165

```

: * TEST 20 <VIA'S ORB/DDR B MASTER CLEAR TEST>
: *
: * ORB == 'OUTPUT REGISTER PORT B'
: * DDRB == 'DATA DIRECTION REGISTER B'
: *
: * FIRST, A MASTER CLEAR IS PERFORMED. NEXT, 377 IS LOADED INTO DDRB TO SET
: * ALL B PORT PINS FOR OUTPUT MODE. THEN, A 000 BYTE IS WRITTEN INTO ORB AND
: * THE REGISTER IS READ BACK AND CHECKED FOR 000. THEN, A MASTER CLEAR IS
: * PERFORMED AND ORB IS READ AND CHECKED FOR 377.
: *****
:
: *****
: * TEST 21 <VIA'S DDR B MASTER CLEAR TEST>
: *
: * DDRB == 'DATA DIRECTION REGISTER B'
: *
: * A 377 BYTE IS WRITTEN INTO DDRB AND THE REGISTER IS READ BACK AND CHECKED
: * FOR 377. THEN, A MASTER CLEAR IS PERFORMED AND DDRB IS READ AND CHECKED FOR
: * 000.
: *
: * NOTE: THIS TESTING IS ALSO DONE IN TEST 23. IT IS INCLUDED HERE ONLY TO
: * PROVIDE TIGHTER LOOPING ON JUST THE DDR B MASTER CLEAR CHECKING.
: *****
:
: *****
: * TEST 22 <VIA'S DDRA MASTER CLEAR TEST>
: *
: * DDRA == 'DATA DIRECTION REGISTER A'
: *
: * A 377 BYTE IS WRITTEN INTO DDRA AND THE REGISTER IS READ BACK AND CHECKED
: * FOR 377. THEN, A MASTER CLEAR IS PERFORMED AND DDRA IS READ AND CHECKED FOR
: * 000.
: *****
:
: *****
: * TEST 23 <VIA'S SR MASTER CLEAR TEST>
: *
: * SR == 'SHIFT REGISTER'
: *
: * A 123 BYTE IS WRITTEN INTO SR AND THE REGISTER IS READ BACK AND CHECKED
: * FOR 123. THEN, A MASTER CLEAR IS PERFORMED AND SR IS READ AND CHECKED FOR
: * NO CHANGE.
: *****
:
: *****
: * TEST 24 <VIA'S ACR MASTER CLEAR TEST>
: *
: * ACR == 'AUXILIARY CONTROL REGISTER'
: *
: * A 252 BYTE IS WRITTEN INTO ACR AND THE REGISTER IS READ BACK AND CHECKED
: * FOR 252. THEN, A MASTER CLEAR IS PERFORMED AND ACR IS READ AND CHECKED FOR
: * 000, TO VERIFY THAT IT IS CLEARED BY MASTER CLEAR.

```



CVDMAA.P11

12-DEC-80 15:59

PROGRAM DOCUMENT

1166  
1167  
1168  
1169  
1170  
1171  
1172  
1173  
1174  
1175  
1176  
1177  
1178  
1179  
1180  
1181  
1182  
1183  
1184  
1185  
1186  
1187  
1188  
1189

\*\*\*\*\*

\*\*\*\*\*

\* TEST 25 <VIA'S PCR MASTER CLEAR TEST>

\* PCR == 'PERIPHERAL CONTROL REGISTER'

\* A 377 BYTE IS WRITTEN INTO PCR AND THE REGISTER IS READ BACK AND CHECKED  
\* FOR 377. THEN, A MASTER CLEAR IS PERFORMED AND PCR IS READ AND CHECKED FOR  
\* 000.

\*\*\*\*\*

\*\*\*\*\*

\* TEST 26 <VIA'S IER MASTER CLEAR TEST>

\* IER == 'INTERRUPT ENABLE REGISTER'

\* A 377 BYTE IS WRITTEN INTO IER AND THE REGISTER IS READ BACK AND CHECKED  
\* FOR 377. THEN, A MASTER CLEAR IS PERFORMED AND IER IS READ AND CHECKED FOR  
\* 200.

\*\*\*\*\*

1190  
 1191  
 1192  
 1193  
 1194  
 1195  
 1196  
 1197  
 1198  
 1199  
 1200  
 1201  
 1202  
 1203  
 1204  
 1205  
 1206  
 1207  
 1208  
 1209  
 1210  
 1211  
 1212  
 1213  
 1214  
 1215  
 1216  
 1217  
 1218  
 1219  
 1220  
 1221  
 1222  
 1223  
 1224  
 1225  
 1226  
 1227  
 1228  
 1229  
 1230  
 1231

8.0 ERROR INFORMATION

8.1 ERROR REPORTING

ERRORS ARE REPORTED BY THE PROGRAM AS THEY OCCUR (IF NOT INHIBITED). THE REPORT CONFORMS TO THE DIAGNOSTIC SUPERVISOR ERROR REPORT FORMAT, AND CONSISTS OF A DESCRIPTION OF THE ERROR, THE TEST NUMBER, SUBTEST NUMBER, PC OF THE ERROR CALL, DEVICE ADDRESS, AND BASIC AND EXTENDED ERROR INFORMATION.

THE FOLLOWING EXAMPLE PROVIDES A TYPICAL ERROR REPORT, WHICH DESCRIBES A 'MASTER CLEAR FAILURE' ERROR, AND PROVIDES THE PC OF THE ERROR CALL AND THE DEVICE REGISTER CONTENTS :

CVDMA DVC FTL ERR 00001 ON UNIT 00 TST 002 SUB 000 PC: 021122  
 MASTER CLEAR FAILURE

THE CONTENTS OF ALL BYTE SELECT REG'S ARE:

BSEL0	BSEL1	BSEL2	BSEL3
000	000	000	000
BSEL4	BSEL5	BSEL6	BSEL7
000	000	121	000
BSEL10	BSEL11	BSEL12	BSEL13
000	000	000	000
BSEL14	BSEL15	BSEL16	BSEL17
000	000	000	000

FOR OTHER ERRORS, THE REPORT MAY BE MORE EXTENSIVE, AND REQUIRE ADDITIONAL DATA TO BE REPORTED.

IF EXTENDED ERROR INFORMATION HAD BEEN INHIBITED USING THE IXE FLAG PRIOR TO RUNNING THE TEST, THE ABOVE ERROR WOULD HAVE BEEN REPORTED IN THE FOLLOWING SHORTENED FORM :

CVDMA DVC FTL ERR 00001 ON UNIT 00 TST 002 SUB 000 PC: 021122  
 MASTER CLEAR FAILURE



CVDMAA.P11 12-DEC-80 15:59

## LISTING &amp; ASSEMBLY CONTROL

```

1232
1233
1234      000000
1235
1236
1237
1238      002000          .=2000
1239
1240
1241 002000          .MCALL SVC
1242
1243 002000          SVC                      ; INITIALIZE SUPERVISOR MACROS
1244
1245
1246          000001          BGNMOD LU1MOD
1247
1248          000001          $LSTIN= 1
1249          000001          $LSTTAG= 1
1250          000001          SVCINS= 1      ; LIST INSTRUCTIONS, SHIFTED RIGHT
1251          000001          SVCTST= 1     ; LIST TEST TAGS, SHIFTED RIGHT
1252          000001          SVCSUB= 1    ; LIST SUBTEST TAGS, SHIFTED RIGHT
1253          000001          SVCGBL= 1    ; LIST GLOBAL TAGS, SHIFTED RIGHT
1254
1255          ; CHANGE THE VALUES OF THE SVC... SYMBOLS TO BE ZERO IF YOU WISH
1256          ; TO ALIGN THE MACRO CALLS AND THEIR EXPANSIONS. CHANGE THE
1257          ; SYMBOLS TO BE MINUS-ONE TO NOT LIST THE EXPANSIONS. YOU MAY
1258          ; CHANGE THE SYMBOLS AT ANY POINT IN YOUR PROGRAM.
1259 002000          POINTER BGNUA,BGNU,ERRTBL
1260

```

CVDMAA.P11 12-DEC-80 15:59

PROGRAM HEADER

```

1261
1262
1263
1264
1265
1266
1267
1268
1269
1270 002000
1271 002000
1272 002000 103
1273 002001 126
1274 002002 104
1275 002003 115
1276 002004 101
1277 002005 000
1278 002006 000
1279 002007 000
1280 002010
1281 002010 101
1282 002011
1283 002011 060
1284 002012
1285 002012 000000
1286 002014
1287 002014 000156
1288 002016
1289 002016 040066
1290 002020
1291 002020 000000
1292 002022
1293 002022 002216
1294 002024
1295 002024 000000
1296 002026
1297 002026 040344
1298 002030
1299 002030 000000
1300 002032
1301 002032 000000
1302 002034
1303 002034 000000
1304 002036
1305 002036 000000
1306 002040
1307 002040 002124
1308 002042
1309 002042 000000
1310 002044
1311 002044 000000
1312 002046
1313 002046 000000
1314 002050
1315 002050 003
1316 002051 003
    
```

.SBTTL PROGRAM HEADER

:++

:THE PROGRAM HEADER MACRO CHARACTERIZES THIS DIAGNOSTIC. THE  
:HEADER MACRO'S ARGUMENTS ARE FILE NAME, RELEASE LEVEL, PATCH  
:DISPOSITION OF THE MOST RECENT PATCH, MAXIMUM TEST TIME IN SEC.,  
:AND THE TYPE OF DIAGNOSTIC (0-SEQUENTIAL, 1-EXERCISER). THESE  
:ARGUMENTS ARE IN RESPECTIVE ORDER.

:--

HEADER CVDMA,A,0,110.,0

```

LSNAME::
        .ASCII /C/
        .ASCII /V/
        .ASCII /D/
        .ASCII /M/
        .ASCII /A/
        .BYTE 0
        .BYTE 0
        .BYTE 0
LSREV::
        .ASCII /A/
LSDEPO::
        .ASCII /0/
LSUNIT::
        .WORD 0
LSTIML::
        .WORD 110.
LSHPCP::
        .WORD LSHARD
LSSPCP::
        .WORD 0
LSHPTP::
        .WORD LSHW
LSSPTP::
        .WORD 0
LSLADP::
        .WORD LSLAST
LSSTA::
        .WORD 0
LSCO::
        .WORD 0
LSDTYP::
        .WORD 0
LSAPT::
        .WORD 0
LSDTP::
        .WORD LSDISPATCH
LSPRIO::
        .WORD 0
LSENV1::
        .WORD 0
LSEXP1::
        .WORD 0
LSMREV::
        .BYTE CSREVISION
        .BYTE CSEDIT
    
```



CVDMAA.P11

12-DEC-80 15:59

PROGRAM HEADER

```

1317 002052
1318 002052 000000
1319 002054 000000
1320 002056
1321 002056 000000
1322 002060
1323 002060 003522
1324 002062
1325 002062 000000
1326 002064
1327 002064 000000
1328 002066
1329 002066 000000
1330 002070
1331 002070 020352
1332 002072
1333 002072 020346
1334 002074
1335 002074 000000
1336 002076
1337 002076 003542
1338 002100
1339 002100 104035
1340 002102
1341 002102 002236
1342 002104
1343 002104 017622
1344 002106
1345 002106 020330
1346 002110
1347 002110 020204
1348 002112
1349 002112 017614
1350 002114
1351 002114 000000
1352 002116
1353 002116 000000
1354 002120
1355 002120 000000
1356
1357

```

.EVEN

```

LSEF:: .WORD 0
        .WORD 0
LSSPC:: .WORD 0
LSDEVP:: .WORD LSDVTYP
LSREPP:: .WORD 0
LSEXP4:: .WORD 0
LSEXP5:: .WORD 0
LSAUT:: .WORD LSAU
LSDUT:: .WORD LSDU
LSLUN:: .WORD 0
LSDESP:: .WORD LSDESC
LSLOAD:: .WORD EMT ESLOAD
LSETP:: .WORD LSERRTBL
LSICP:: .WORD LSINIT
LSCCP:: .WORD LSCLEAN
LSACP:: .WORD LSAUTO
LSPRT:: .WORD LSPROT
LSTEST:: .WORD 0
LSDLY:: .WORD 0
LSHIME:: .WORD 0

```

CVDMAA.P11 12-DEC-80 15:59

DISPATCH TABLE

.SBTTL DISPATCH TABLE

```

:////////////////////
:// THE DISPATCH TABLE CONTAINS THE STARTING ADDRESS OF EACH TEST.
:// IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.
:////////////////////

```

```

1358
1359
1360
1361
1362
1363
1364
1365 002122
1366 002122 000034
1367 002124
1368 002124 020354
1369 002126 021120
1370 002130 021230
1371 002132 021422
1372 002134 021564
1373 002136 021706
1374 002140 022110
1375 002142 022620
1376 002144 023166
1377 002146 024134
1378 002150 025114
1379 002152 025736
1380 002154 026020
1381 002156 026102
1382 002160 026204
1383 002162 027250
1384 002164 027332
1385 002166 027414
1386 002170 027500
1387 002172 027566
1388 002174 030054
1389 002176 030204
1390 002200 030334
1391 002202 030470
1392 002204 030620
1393 002206 030750
1394 002210 031106
1395 002212 036332
1396

```

DISPATCH 28.

```

          .WORD 28
LSDISPATCH::
          .WORD T1
          .WORD T2
          .WORD T3
          .WORD T4
          .WORD T5
          .WORD T6
          .WORD T7
          .WORD T8
          .WORD T9
          .WORD T10
          .WORD T11
          .WORD T12
          .WORD T13
          .WORD T14
          .WORD T15
          .WORD T16
          .WORD T17
          .WORD T18
          .WORD T19
          .WORD T20
          .WORD T21
          .WORD T22
          .WORD T23
          .WORD T24
          .WORD T25
          .WORD T26
          .WORD T27
          .WORD T28

```



CVDMAA.P11 12-DEC-80 15:59

DEFAULT HARDWARE P-TABLE

.SBTTL DEFAULT HARDWARE P-TABLE

```

:////////////////////
:/ THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF
:/ THE TEST-DEVICE PARAMETERS. THE STRUCTURE OF THIS TABLE
:/ IS IDENTICAL TO THE STRUCTURE OF THE RUN-TIME P-TABLE.
:////////////////////

```

```

1397
1398
1399
1400
1401
1402
1403
1404
1405 002214
1406 002214 000007
1407 002216
1408 002216
1409
1410 002216 160020
1411 002220 000300
1412 002222 004000
1413 002224 000000
1414 002226 000000
1415 002230 000000
1416 002232 000111
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427 002234
1428 002234

```

BGNHW DFPTBL

```

.LSHW:: .WORD L10000-LSHW/2
DFPTBL::

```

```

.WORD 160020 :DMV11 CSR UNIBUS ADDRESS
.WORD 300 :DMV11 INTERRUPT VECTOR
.WORD 4000 :DMV11 INTERRUPT PRIORITY LEVEL = 4
.WORD 000 :SWITCH REG. #1 (BOOT ADDRESS)
.WORD 000 :SWITCH REG. #2 (DDCMP ADDRESS)
.WORD 0 :H3254&H3255 USED
.WORD 000111 :MISC. CONTROLS:

```

```

: POWER-UP MODE 0 MASK = 100
: 0 = NOT JUMPERED FOR MODE 0 POWER-UP
: 1 = JUMPERED FOR MODE 0 POWER-UP <=== DEFAULT SETTING
: BOTH W5 & W6 REMOVED

```

```

: BAUD RATE MASK = 77
: 7 = 19.2 K
: 11 = 56 K <=== DEFAULT SETTING

```

ENDHW

L10000:

CVDMAA.P11 12-DEC-80 15:59

SOFTWARE P-TABLE

1429  
1430  
1431  
1432  
1433  
1434  
1435  
1436  
1437  
1438  
1439  
1440  
1441

002234  
002234 000000  
002236  
002236  
002236  
002236

```
.SBTTL SOFTWARE P-TABLE  
://////  
: THE SOFTWARE P-TABLE CONTAINS THE VALUES OF THE PROGRAM  
: PARAMETERS THAT CAN BE CHANGED BY THE OPERATOR.  
://////
```

BGNSW SFPTBL

ENDSW

```
.WORD L10001-LSSW/2  
LSSW::  
SFPTBL::  
L10001:
```



CVDMAA.P11 12-DEC-80 15:59

GLOBAL EQUATES SECTION

.SBTTL GLOBAL EQUATES SECTION

```

://////
:/ THE GLOBAL EQUATES SECTION CONTAINS PROGRAM EQUATES THAT
:/ ARE USED IN MORE THAN ONE TEST.
://////

```

EQUALS

: BIT DIFINITIONS

1442		
1443		
1444		
1445		
1446		
1447		
1448		
1449		
1450	002236	
1451		
1452		
1453		
1454	100000	BIT15== 100000
1455	040000	BIT14== 40000
1456	020000	BIT13== 20000
1457	010000	BIT12== 10000
1458	004000	BIT11== 4000
1459	002000	BIT10== 2000
1460	001000	BIT09== 1000
1461	000400	BIT08== 400
1462	000200	BIT07== 200
1463	000100	BIT06== 100
1464	000040	BIT05== 40
1465	000020	BIT04== 20
1466	000010	BIT03== 10
1467	000004	BIT02== 4
1468	000002	BIT01== 2
1469	000001	BIT00== 1

```

:
: BIT9== BIT09
: BIT8== BIT08
: BIT7== BIT07
: BIT6== BIT06
: BIT5== BIT05
: BIT4== BIT04
: BIT3== BIT03
: BIT2== BIT02
: BIT1== BIT01
: BIT0== BIT00

```

```

: EVENT FLAG DEFINITIONS
: EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION

```

1485	000040	EF.START== 32.	: START COMMAND WAS ISSUED
1486	000037	EF.RESTART== 31.	: RESTART COMMAND WAS ISSUED
1487	000036	EF.CONTINUE== 30.	: CONTINUE COMMAND WAS ISSUED
1488	000035	EF.NEW== 29.	: A NEW PASS HAS BEEN STARTED
1489	000034	EF.PWR== 28.	: A POWER-FAIL/POWER-UP OCCURRED

: PRIORITY LEVEL DEFINITIONS

1493		
1494	000340	PRI07== 340
1495	000300	PRI06== 300
1496	000240	PRI05== 240
1497	000200	PRI04== 200

CVDMAA.P11

12-DEC-80 15:59

## GLOBAL EQUATES SECTION

```

1498      000140      PRI03== 140
1499      000100      PRI02== 100
1500      000040      PRI01== 40
1501      000000      PRI00== 0
1502
1503      ;OPERATOR FLAG BITS
1504
1505      000004      EVL==      4
1506      000010      LOT==      10
1507      000020      ADR==      20
1508      000040      IDU==      40
1509      000100      ISR==     100
1510      000200      UAM==     200
1511      000400      BOE==     400
1512      001000      PNT==    1000
1513      002000      PRI==    2000
1514      004000      IXE==    4000
1515      010000      IBE==   10000
1516      020000      IER==   20000
1517      040000      LOE==   40000
1518      100000      HOE==  100000
1519
1520      .SBTTL DEFINE THE NUMBER OF CSR'S
1521      000020      CSREGS = 16.
1522
1523      ;-----
1524
1525      .SBTTL NPR ADDRESS REGISTER EQUATES
1526      000070      NPRAOL = 70      ;OUT NPR ADRS LO REG
1527      000071      NPRAOH = NPRAOL+1 ;OUT NPR ADRS HI REG
1528      000072      NPRAOX = NPRAOL+2 ;OUT NPR EXTENDED ADRS REG
1529      000074      NPRAIL = NPRAOL+4 ;IN NPR ADRS LO REG
1530      000075      NPRAIH = NPRAOL+5 ;IN NPR ADRS HI REG
1531      000076      NPRAIX = NPRAOL+6 ;IN NPR EXTENDED ADRS REG
1532      000010      NPRBS7 = BIT3    ;'BANK SELECT 7' BIT -- W/IN EXTENDED ADRS. REG.
1533
1534
1535
1536      .SBTTL NPR DATA REG EQUATES
1537      123000      NPRDRL = 123000 ;NPR DATA REGISTER -- LOW BYTE
1538      123001      NPRDRH = NPRDRL+1 ;NPR DATA REGISTER -- HIGH BYTE
1539
1540
1541
1542      .SBTTL NPR CONTROL REG EQUATES
1543      123004      NPRCTL = NPRDRL+4 ;NPR CONTROL REGISTER
1544      000200      NPRABT = BIT7    ;=1 IF BUS TIME-OUT ON NPR
1545      000100      NPRGO  = BIT6    ;SET FOR NOP, CLEAR TO 'GO' / 0=DONE, 1=BUSY
1546      000040      NPRIO  = BIT5    ;0 = (LSI ==> DMV); 1 = (DMV ==> LSI)
1547      000020      LSIHLT = BIT4    ;SETTING THIS WILL 'HALT' THE LSI-11 !!
1548      000010      NPRBYT = BIT3    ;SET TO 1 TO WRITE BYTE ONLY TO LSI-11
1549      000004      DMVPU  = BIT2    ;SET BY MICRO-DIAG. MUST REMAIN SET!!!
1550      000002      LSIACL = BIT1    ;IF SET, WILL CAUSE POWER DOWN CONDITION IN LSI!
1551      000001      DMVDAI = BIT0    ;'DISABLE INIT' FROM EFFECTING DMV-11
1552
1553

```



CVDMAA.P11 12-DEC-80 15:59

## NPR REQUEST FUNCTIONS

```

1554
1555      000004
1556      000044
1557      000054
1558
1559
1560
1561
1562      123005
1563      000004
1564      000002
1565
1566
1567
1568
1569      000001
1570

```

```

.SBTTL  NPR REQUEST FUNCTIONS
NPRDL   = DMVPU           ;WORD XFER:  LSI ==> DMV
NPRDL   = DMVPU!NPRIO     ;WORD XFER:  DMV ==> LSI
NPRDLB  = DMVPU!NPRIO!NPRBYT ;BYTE XFER:  DMV ==> LSI
;-----
.SBTTL  INTERRUPT REG EQUATES
IRQREG  = 123005         ;INTERRUPT REQUEST REG
IRQA    = BIT2           ;REQUEST BIT FOR XX0 INTERRUPT -- 'A'
IRQB    = BIT1           ;REQUEST BIT FOR XX4 INTERRUPT -- 'B'
;-----
.SBTTL  CONTROL FLAGS FROM P-TABLE ENTRIES
PU24    = BIT0           ;POWER-FAIL VECTORING MODE. 1 = MODE 0
; (I.E. JUMPERS W5 & W6 BOTH REMOVED)

```

CVDMAA.P11

12-DEC-80 15:59

SWITCH PACKS

1571  
1572  
1573  
1574  
1575  
1576  
1577  
1578  
1579

121000  
121400

.SBTTL SWITCH PACKS

::\*\*\*\*\*  
:\* SWITCH PACKS  
:\*\*\*\*\*

SWPBOT = 121000  
SWPDDCMP = 121400

;'BOOT ADDRESS'' SWITCH PACK [A200]  
;'DDCMP ADDRESS'' SWITCH PACK [A300]



CVDMAA.P11

12-DEC-80 15:59

CSR REG. DEFINITION FOR MAINT. LOOP

1580  
1581  
1582  
1583  
1584  
1585  
1586  
1587  
1588  
1589  
1590  
1591  
1592  
1593  
1594  
1595  
1596  
1597  
1598  
1599  
1600  
1601  
1602  
1603  
1604  
1605  
1606  
1607  
1608  
1609  
1610  
1611  
1612

000001  
000020

000200  
000100  
000001

000001  
000002  
000003  
000004  
000005  
000200

.SBTTL CSR REG. DEFINITION FOR MAINT. LOOP

:+\*\*\*\*\*

.SBTTL MAINTENANCE REGISTER - BSEL0

:--\*\*\*\*\*

: INTERRUPT ENABLE BITS

IENBA = BIT0 ;INTERRUPT ENABLE 'A'  
IENBB = BIT4 ;INTERRUPT ENABLE 'B'

:+\*\*\*\*\*

.SBTTL MAINTENANCE REGISTER - BSEL1

:--\*\*\*\*\*

: MAINT. LOOP CONTROL BITS:

RUN = BIT7  
MCLR = BIT6  
MREQ = BIT0

:+\*\*\*\*\*

.SBTTL MAINTENANCE REGISTER - BSEL2

:--\*\*\*\*\*

: MAINTENANCE FUNCTION CODES

REDLOC = 1 ;FUNCTION CODE FOR READ A 6502 LOCATION  
WRILOC = 2 ;FUNCTION CODE FOR WRITE A 6502 LOCATION  
REDPAG = 3 ;FUNCTION CODE FOR READ A 6502 MEMORY PAGE  
WRIPAG = 4 ;FUNCTION CODE FOR WRITE A 6502 RAM PAGE  
EXECUT = 5 ;FUNCTION CODE FOR EXECUTE AT GIVEN PC

MRDY = BIT7 ;M-LOOP REDY FOR A COMMAND WHEN SET

CVDMAA.P11 12-DEC-80 15:59

DMV INTERNAL ADDRESSES

.SBTTL DMV INTERNAL ADDRESSES

```

:++*****
:
:          DMV INTERNAL ADDRESSES
:--*****
    
```

:\*\*\*\*\* << MICROPROCESSOR REGISTER ADDRESS EQUATES >> \*\*\*\*\*

.SBTTL BYTE & WORD SELECT REGISTERS

1624	000020	SLT0	=020
1625	000020	BSLT0	=SLT0
1626	000021	BSLT1	=SLT0+1
1627	000022	SLT2	=SLT0+2
1628	000022	BSLT2	=SLT0+2
1629	000023	BSLT3	=SLT0+3
1630	000024	SLT4	=SLT0+4
1631	000024	BSLT4	=SLT0+4
1632	000025	BSLT5	=SLT0+5
1633	000026	SLT6	=SLT0+6
1634	000026	BSLT6	=SLT0+6
1635	000027	BSLT7	=SLT0+7

.SBTTL VIA'S REGISTERS

1639	120000	ORB	=120000
1640	120001	ORA	=ORB+1
1641	120002	DDRB	=ORB+2
1642	120003	DDRA	=ORB+3
1643	120004	T1CL	=ORB+4
1644	120005	T1CH	=ORB+5
1645	120005	T1LHGO	=ORB+5
1646	120006	T1LL	=ORB+6
1647	120007	T1LH	=ORB+7
1648	120010	T2LL	=ORB+10
1649	120010	T2CL	=T2LL
1650	120011	T2CH	=ORB+11
1651	120012	SR	=ORB+12
1652	120013	ACR	=ORB+13
1653	120014	PCR	=ORB+14
1654	120015	IFR	=ORB+15
1655	120016	IENR	=ORB+16
1656	120017	ORAM	=ORB+17

.SBTTL VIA'S "IFR" REGISTER'S BIT ASSIGNMENTS

1660	000200	IFRIRQ	=BIT7	: "IRQ" HAS BEEN ISSUED -- LOGICAL 'OR' OF BITS 0 --> 6
1661	000100	IFRT1	=BIT6	: "T1" -- TIMER # 1 TIMED-OUT
1662	000040	IFRT2	=BIT5	: "T2" -- TIMER # 1 TIMED-OUT
1663	000020	IFRCB1	=BIT4	: "CB1" EDGE DETECTED ('K2 LINE UNIT STEP' O/P SIGNAL FROM SR)
1664	000010	IFRCB2	=BIT3	: "CB2" EDGE DETECTED (UNUSED!)
1665	000004	IFRSR	=BIT2	: "SR" REGISTER COMPLETED SHIFT OPERATION
1666	000002	IFRCA1	=BIT1	: "CA1" EDGE DETECTED ('K6 MOD RDY H')
1667	000001	IFRCA2	=BIT0	: "CA2" EDGE DETECTED ('K2 CTS H')
1668				



CVDMAA.P11 12-DEC-80 15:59

VIA'S "IFR" REGISTER'S BIT ASSIGNMENTS

1669

CVDMAA.P11 12-DEC-80 15:59

GLOBAL DATA SECTION

.SBTTL GLOBAL DATA SECTION

```

:////////////////////
:/ THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
:/ IN MORE THAN ONE TEST.
:////////////////////

```

```

:++*****
.SBTTL CONTROL BLOCK FOR STACKED ERROR MESSAGES
:--*****

```

ERRTBL

L\$ERRTBL::

```

ERRTYP:: .WORD 0
ERRNBR:: .WORD 0
ERRMSG:: .WORD 0
ERRBLK:: .WORD 0

```

```

:++*****
.SBTTL STORAGE FOR DEVICE REGISTERS
:--*****

```

```

WSR0:
BSR0: .WORD 0
WSR2:
BSR1: .WORD 0
WSR4:
BSR2: .WORD 0
WSR6:
BSR3: .WORD 0
WSR10:
BSR4: .WORD 0
WSR12:
BSR5: .WORD 0
WSR14:
BSR6: .WORD 0
WSR16:
BSR7: .WORD 0
BSR10: .WORD 0
BSR11: .WORD 0
BSR12: .WORD 0
BSR13: .WORD 0
BSR14: .WORD 0
BSR15: .WORD 0
BSR16: .WORD 0
BSR17: .WORD 0

```

```

:++*****
.SBTTL MISCELLANEOUS STORAGE
:--*****

```

```

TDATA: .WORD 0 ;TEST DATA
GDATA: .WORD 0 ;EXPECTED DATA
BDATA: .WORD 0 ;ACTUAL DATA
XDATA: .WORD 0 ;EXCLUSIVE OR BETWEEN 'GDATA' & 'BDATA'
DELAY1: .WORD 110400 ;DELAY TIME, 3 INST., 500 MILLISEC.
DELAY2: .WORD 7 ;DELAY TIME FOR M-LOOP FUNCTION, 100 USEC.APPROX.
LOGDEV: .WORD 0 ;LOGICAL DEVICE NUMBER

```

```

1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681 002236
1682 002236
1683 002236 000000
1684 002240 000000
1685 002242 000000
1686 002244 000000
1687
1688
1689
1690
1691 002246
1692 002246 000000
1693 002250
1694 002250 000000
1695 002252
1696 002252 000000
1697 002254
1698 002254 000000
1699 002256
1700 002256 000000
1701 002260
1702 002260 000000
1703 002262
1704 002262 000000
1705 002264
1706 002264 000000
1707 002266 000000
1708 002270 000000
1709 002272 000000
1710 002274 000000
1711 002276 000000
1712 002300 000000
1713 002302 000000
1714 002304 000000
1715
1716
1717
1718
1719 002306 000000
1720 002310 000000
1721 002312 000000
1722 002314 000000
1723 002316 110400
1724 002320 000007
1725 002322 000000

```



CVDMAA.P11 12-DEC-80 15:59

## MISCELLANEOUS STORAGE

1726	002324	000000	PSTACK: .WORD	0	:CONTAINS BASE LEVEL PROGRAM STACK POINTER
1727	002326	000000	INTFLG: .WORD	0	: INTERRUPT RECEIVED FLAG BYTES. ALLOCATION:
1728					: LOW BYTE FOR 'A' & HIGH BYTE FOR 'B'
1729	002330	000000	INTWCH: .WORD	0	: BYTE IS SET NON-ZERO WHEN HANDLER SHOULD BE
1730					: WATCHING FOR INT'S. ALLOCATION: SEE INTFLG
1731	002332	000000	ERRFLG: .WORD	0	: ERROR FLAG
1732	002334	000000	REGNUM: .WORD	0	: REGISTER NUMBER -- FOR PASSING ARG. TO 'ERR#'
1733	002336	000000	FRSTIM: .WORD	0	: FLAG=0 IF PROGRAM JUST LOADED
1734	002340	000000	FRSPAS: .WORD	0	: FLAG=0 IF FIRST PASS AFTER LOAD
1735	002342	000000	DEVMAP: .WORD	0	: BIT MAP OF ACTIVE DEVICES
1736	002344	000000	DEVPTR: .WORD	0	: DEVICE MAP BIT POINTER
1737	002346	000000	CONSOL: .WORD	0	: CONSOLE DEVICE FLAG -- NON-ZERO = NONE PRESENT
1738	002350	000000	PFLAG: .WORD	0	: MISC. PROGRAM FLAGS
1739					
1740					: THE ABOVE WORD CONTAINS MISC. FLAGS WHICH CAN ONLY BE ACCESSED BY PATCHING.
1741					: IT IS NOT INTENDED THAT THEY BE SET OR CLEARED EXCEPT UNDER VERY UNUSUAL
1742					: CIRCUMSTANCES. THEREFORE, THEY WILL NOT BE DOCUMENTED ANY OTHER PLACE
1743					: EXCEPT RIGHT HERE.
1744					
1745					: BIT 0 -- WHEN SET, THOSE TESTS WHICH DO A BUS RESET WILL NOT BE EXECUTED.
1746					: THIS WAS IMPLEMENTED TO SAVE WEAR & TEAR ON THE RX01 IN THE
1747					: DEVELOPMENT SYSTEM WHILE DOING LONG TERM TESTING OF ALL OTHER
1748					: TESTS.
1749					
1750					: BIT 1 -- CPU TYPE. (NOT USED)
1751					
1752					: BIT 2 -- CONTROLS PRINTING OF EXTENDED ERROR INFORMATION DURING 'MOVING
1753					: INVERSIONS TEST' OF RAM. NORMALLY ONLY ADDRESS, GOOD & BAD
1754					: DATA, AND XOR WILL BE PRINTED. IF THIS BIT IS SET HOWEVER,
1755					: INFORMATION IDENTIFYING WHERE WITHIN THE ALGORITHM THE ERROR
1756					: WAS DETECTED IS REPORTED. THE FOLLOWING ABBREVIATIONS ARE USED
1757					: IN THE HEADING:
1758					: BIT --- IDENTIFIES THE INNERMOST LOOP. WHICH BIT IS
1759					: BEING INVERTED AT EACH LOCATION. BITS ARE
1760					: IDENTIFIED AS 0 THROUGH 7.
1761					: DATA --- IDENTIFIES THE VALUE TO WHICH THE ABOVE BIT IS
1762					: BEING SET (I.E. 0 OR 1). IT IS FIRST READ AND
1763					: CHECKED FOR EXPECTED CONTENTS; THEN THE BIT IS
1764					: INVERTED TO THIS STATE (DATA) AND RE-WRITTEN;
1765					: THEN IT IS AGAIN READ & CHECKED FOR THE NEW
1766					: VALUE.
1767					: SEQ --- INDICATES THE DIRECTION (FWD OR BKWD) THE TEST
1768					: WAS SCANNING THROUGH RAM WHEN THE ERROR OCCURED.
1769					: LSB --- THIS IS THE LOGICAL LEAST SIGNIFICANT BIT OF THE
1770					: RAM ADDRESS AS WE SCAN THROUGH MEMORY. BY
1771					: VARYING THIS, THE ALGORITHM GENERATS NON-SEQUEN-
1772					: TIAL ADDRESSING OF RAM AND EFFECTS A MUCH MORE
1773					: THOROUGH TEST OF MEMORY.
1774					
1775					
1776					

CVDMAA.P11 12-DEC-80 15:59

## CURRENT DEVICE PARAMETERS

```

1777
1778
1779          160000
1780
1781 002352
1782 002352
1783 002352
1784 002352 160000
1785 002354 160001
1786 002356
1787 002356 160002
1788 002360 160003
1789 002362
1790 002362 160004
1791 002364 160005
1792 002366
1793 002366 160006
1794 002370 160007
1795 002372
1796 002372 160010
1797 002374 160011
1798 002376
1799 002376 160012
1800 002400 160013
1801 002402
1802 002402 160014
1803 002404 160015
1804 002406
1805 002406 160016
1806 002410 160017
1807
1808 002412 000300
1809 002414 000304
1810 002416 000340
1811
1812
1813
1814 002420 000000
1815 002422 000000
1816 002424 000000
1817 002426 000000
1818 002430 000000
1819 002432 000000
1820 002434 000000
1821 002436 000000
1822

```

```

CURRENT DEVICE PARAMETERS
.SBTTL CURRENT DEVICE PARAMETERS
$MPCSR == 160000 ;INITIAL ASSEMBLED IN CSR ADDRESS
MPCSR: ;POINTER TO THE DMV11 CSR'S
BSEL0: ;POINTER TO BSEL0
BSEL: ;ALTERNATE NAME FOR BSEL0
SEL0: .WORD $MPCSR ;POINTER TO SEL0
BSEL1: .WORD $MPCSR+1 ;POINTER TO BSEL1
BSEL2: ;POINTER TO BSEL2
SEL2: .WORD $MPCSR+2 ;POINTER TO SEL2
BSEL3: .WORD $MPCSR+3 ;POINTER TO BSEL3
BSEL4: ;POINTER TO BSEL4
SEL4: .WORD $MPCSR+4 ;POINTER TO SEL4
BSEL5: .WORD $MPCSR+5 ;POINTER TO BSEL5
BSEL6: ;POINTER TO BSEL6
SEL6: .WORD $MPCSR+6 ;POINTER TO SEL6
BSEL7: .WORD $MPCSR+7 ;POINTER TO BSEL7
BSEL10: ;POINTER TO BSEL10
SEL10: .WORD $MPCSR+10 ;POINTER TO SEL10
BSEL11: .WORD $MPCSR+11 ;POINTER TO BSEL11
BSEL12: ;POINTER TO BSEL12
SEL12: .WORD $MPCSR+12 ;POINTER TO SEL12
BSEL13: .WORD $MPCSR+13 ;POINTER TO BSEL13
BSEL14: ;POINTER TO BSEL14
SEL14: .WORD $MPCSR+14 ;POINTER TO SEL14
BSEL15: .WORD $MPCSR+15 ;POINTER TO BSEL15
BSEL16: ;POINTER TO BSEL16
SEL16: .WORD $MPCSR+16 ;POINTER TO SEL16
BSEL17: .WORD $MPCSR+17 ;POINTER TO BSEL17
MPIVEC: .WORD 300 ;DMV11 INPUT INTERRUPT VECTOR
MPOVEC: .WORD 304 ;DMV11 OUTPUT INTERRUPT VECTOR
MPRIOR: .WORD 340 ;DMV11 DEVICE PRIORITY
.SBTTL GEN'L PURPOSE SCRATCH STORAGE
REG0: .WORD 0
REG1: .WORD 0
REG2: .WORD 0
REG3: .WORD 0
REG4: .WORD 0
REG5: .WORD 0
REG6: .WORD 0
REG7: .WORD 0

```



CVDMAA.P11 12-DEC-80 15:59

SCRATCH STORAGE FOR MESSAGE REPORTING

1823  
 1824  
 1825 002440 000000  
 1826 002442 000000  
 1827 002444 000000  
 1828 002446 000000  
 1829 002450 000000  
 1830 002452 000000  
 1831 002454 000000  
 1832 002456 000000  
 1833 002460 000000  
 1834 002462 000000  
 1835 002464 000000  
 1836 002466 000000  
 1837 002470 000000  
 1838 002472 000000  
 1839 002474 000000  
 1840 002476 000000  
 1841 002500 000000  
 1842 002502 000000  
 1843  
 1844

.SBTTL SCRATCH STORAGE FOR MESSAGE REPORTING

TMP0: .WORD 0  
 TMP1: .WORD 0  
 TMP2: .WORD 0  
 TMP3: .WORD 0  
 TMP4: .WORD 0  
 TMP5: .WORD 0  
 TMP6: .WORD 0  
 TMP7: .WORD 0  
 TMP8: .WORD 0  
 TMP9: .WORD 0  
 TMPA: .WORD 0  
 TMPB: .WORD 0  
 TMPC: .WORD 0  
 TMPD: .WORD 0  
 TMPE: .WORD 0  
 TMPF: .WORD 0  
 NEWPC: .WORD 0  
 OLDSP: .WORD 0

:SAVE LOCATION FOR A 'PC' VALUE RESET  
 :SAVE LOCATION FOR A STACK POINTER RESET VALUE

CVDMAA.P11 12-DEC-80 15:59

## \*\*\*\*\* DATA PATTERN A \*\*\*\*\*

## .SBTTL \*\*\*\*\* DATA PATTERN A \*\*\*\*\*

.EVEN	.WORD	PATB-PATA-2	:USAGE:
PATA:	.BYTE	001	:# OF BYTES IN PATTERN
	.BYTE	002	:BSEL0
	.BYTE	004	:BSEL1
	.BYTE	010	:BSEL2
	.BYTE	020	:BSEL3
	.BYTE	040	:BSEL4
	.BYTE	100	:BSEL5
	.BYTE	200	:BSEL6
	.BYTE	052	:BSEL7
	.BYTE	300	:BSEL10
	.BYTE	140	:BSEL11
	.BYTE	060	:BSEL12
	.BYTE	030	:BSEL13
	.BYTE	014	:BSEL14
	.BYTE	006	:BSEL15
	.BYTE	003	:BSEL16
	.BYTE		:BSEL17

## .SBTTL \*\*\*\*\* DATA PATTERN B \*\*\*\*\*

.EVEN	.WORD	PATC-PATB-2	:USAGE:
PATB:	.BYTE	125	:# OF BYTES IN PATTERN
	.BYTE	252	
	.BYTE	000	
	.BYTE	377	
	.BYTE	001	
	.BYTE	002	
	.BYTE	004	
	.BYTE	010	
	.BYTE	020	
	.BYTE	040	
	.BYTE	100	
	.BYTE	200	
	.BYTE	376	
	.BYTE	375	
	.BYTE	373	
	.BYTE	367	
	.BYTE	357	
	.BYTE	337	
	.BYTE	277	
	.BYTE	177	
	.BYTE	000	

```

1845
1846
1847
1848 002504 000020
1849 002506 001
1850 002507 002
1851 002510 004
1852 002511 010
1853 002512 020
1854 002513 040
1855 002514 100
1856 002515 200
1857 002516 052
1858 002517 300
1859 002520 140
1860 002521 060
1861 002522 030
1862 002523 014
1863 002524 006
1864 002525 003
1865
1866
1867
1868
1869 002526 000026
1870 002530 125
1871 002531 252
1872 002532 000
1873 002533 377
1874 002534 001
1875 002535 002
1876 002536 004
1877 002537 010
1878 002540 020
1879 002541 040
1880 002542 100
1881 002543 200
1882 002544 376
1883 002545 375
1884 002546 373
1885 002547 367
1886 002550 357
1887 002551 337
1888 002552 277
1889 002553 177
1890 002554 000
1891

```



CVDMAA.P11 12-DEC-80 15:59

\*\*\*\*\* DATA PATTERN C \*\*\*\*\*

```

1892
1893
1894
1895
1896      002556      002556
1897      002556      000012      377
1898      002560      002      377
1899      002562      003      366
1900      002564      000      100
1901      002566      013      040
1902      002570      006      106
1903      002572      007      107
1904      002574      012      112
1905      002576      014      042
1906      002600      015      000
1907      002602      016      200
1908
1909
1910
1911      002604      100
1912      002605      000
1913      002606      377
1914      002607      366
1915      002610      000
1916      002611      000
1917      002612      106
1918      002613      107
1919      002614      000
1920      002615      000
1921      002616      112
1922      002617      040
1923      002620      042
1924      002621      000
1925      002622      200
1926      002623      000
1927
1928
1929
1930
1931
1932      002624      000
1933      002625      377
1934      002626      000
1935      002627      000
1936      002630      377
1937      002631      377
1938      002632      000
1939      002633      000
1940      002634      377
1941      002635      377
1942      002636      000
1943      002637      000
1944      002640      000
1945      002641      377
1946      002642      200
1947      002643      377
    
```

.SBTTL \*\*\*\*\* DATA PATTERN C \*\*\*\*\*

USED BY TEST # 11 TO LOAD UP THE VIA'S REGISTERS. THE REGISTER NUMBER LOADED IS THE FIRST BYTE AND THE VALUE LOADED INTO IT IS THE SECOND BYTE

.EVEN

PATC: .WORD <PATCR-PATC-2>/2

```

.BYTE 2,377 :SETUP ORB AS AN I/O (READ/WRITE) REGISTER
.BYTE 3,366 :SETUP ORA AS AN O/P REGISTER -- IT CAN'T BE TESTED!
.BYTE 0,100 :LOAD UP ORB
.BYTE 13,040 :ACR
.BYTE 6,106 :T1LL
.BYTE 7,107 :T1LH
.BYTE 12,112 :SR
.BYTE 14,042 :PCR
.BYTE 15,000 :IFR
.BYTE 16,200 :IER
    
```

: THIS TABLE IS THE LIST OF EXPECTED CONTENTS OF THE VIA'S REGISTERS

```

PATCR: .BYTE 100 : ORB
       .BYTE 000 : ORA
       .BYTE 377 : DDRB
       .BYTE 366 : DDRA
       .BYTE 000 : T1CL
       .BYTE 000 : T1CH
       .BYTE 106 : T1LL
       .BYTE 107 : T1LH
       .BYTE 000 : T2CL
       .BYTE 000 : T2CH
       .BYTE 112 : SR
       .BYTE 040 : ACR
       .BYTE 042 : PCR
       .BYTE 000 : IFR
       .BYTE 200 : IER
       .BYTE 000 : ORA
    
```

: THIS IS THE TABLE OF TEST PATTERN 'A' MASKS. BEFORE A REGISTER'S CONTENTS IS TESTED, A BICB IS DONE USING ITS RESPECTIVE BYTE FROM THE TABLE BELOW (INSURING THAT 'DON'T CARE' BITS ARE IGNORED).

```

PATCM: .BYTE 000 : ORB
       .BYTE 377 : ORA -- THIS REGISTER CAN'T BE TESTED!!!
       .BYTE 000 : DDRB
       .BYTE 000 : DDRA
       .BYTE 377 : T1CL -- THIS IS A FREE RUNNING COUNTER
       .BYTE 377 : T1CH -- THIS IS A FREE RUNNING COUNTER
       .BYTE 000 : T1LL
       .BYTE 000 : T1LH
       .BYTE 377 : T2CL -- THIS IS A FREE RUNNING COUNTER
       .BYTE 377 : T2CH -- THIS IS A FREE RUNNING COUNTER
       .BYTE 000 : SR
       .BYTE 000 : ACR
       .BYTE 000 : PCR
       .BYTE 377 : IFR
       .BYTE 200 : IER -- BIT 7 IS ALWAYS READ AS ZERO
       .BYTE 377 : ORA -- THIS REGISTER CAN'T BE TESTED!!!
    
```

CVDMAA.P11 12-DEC-80 15:59

\*\*\*\*\* DATA PATTERN D \*\*\*\*\*

.SBTTL \*\*\*\*\* DATA PATTERN D \*\*\*\*\*

```

.EVEN
PATD:  .WORD  PATE-PATD-2
        .BYTE  200
        .BYTE  201
        .BYTE  202
        .BYTE  204
        .BYTE  210
        .BYTE  220
        .BYTE  240
        .BYTE  300
        .BYTE  200
        .BYTE  000
        .BYTE  001
        .BYTE  002
        .BYTE  004
        .BYTE  010
        .BYTE  020
        .BYTE  040
        .BYTE  100
        .BYTE  000
        .BYTE  325
        .BYTE  125
        .BYTE  252
        .BYTE  052

```

```

1948
1949
1950
1951 002644 000026
1952 002646 200
1953 002647 201
1954 002650 202
1955 002651 204
1956 002652 210
1957 002653 220
1958 002654 240
1959 002655 300
1960 002656 200
1961 002657 000
1962 002660 001
1963 002661 002
1964 002662 004
1965 002663 010
1966 002664 020
1967 002665 040
1968 002666 100
1969 002667 000
1970 002670 325
1971 002671 125
1972 002672 252
1973 002673 052
1974
1975
1976
1977
1978

```

.SBTTL \*\*\*\*\* DATA PATTERN E \*\*\*\*\*

```

.EVEN
PATE:  .WORD  PATF-PATE-2
        .BYTE  200
        .BYTE  201
        .BYTE  203
        .BYTE  207
        .BYTE  217
        .BYTE  237
        .BYTE  277
        .BYTE  377
        .BYTE  377
        .BYTE  377
        .BYTE  377
        .BYTE  376
        .BYTE  374
        .BYTE  370
        .BYTE  360
        .BYTE  340
        .BYTE  300
        .BYTE  200
        .BYTE  200
        .BYTE  200
        .BYTE  325
        .BYTE  200
        .BYTE  252
        .BYTE  200

```

```

1979 002674 000026
1980 002676 200
1981 002677 201
1982 002700 203
1983 002701 207
1984 002702 217
1985 002703 237
1986 002704 277
1987 002705 377
1988 002706 377
1989 002707 377
1990 002710 376
1991 002711 374
1992 002712 370
1993 002713 360
1994 002714 340
1995 002715 300
1996 002716 200
1997 002717 200
1998 002720 325
1999 002721 200
2000 002722 252
2001 002723 200

```



CVDMAA.P11 12-DEC-80 15:59

\*\*\*\*\* DATA PATTERN F \*\*\*\*\*

.SBTTL \*\*\*\*\* DATA PATTERN F \*\*\*\*\*

.EVEN  
PATF: .WORD <PATG-PATF-2>/2

2002		
2003		
2004		
2005	002724	000045
2006	002726	125252
2007	002730	052525
2008	002732	000000
2009	002734	177777
2010	002736	000001
2011	002740	000002
2012	002742	000004
2013	002744	000010
2014	002746	000020
2015	002750	000040
2016	002752	000100
2017	002754	000200
2018	002756	000400
2019	002760	001000
2020	002762	002000
2021	002764	004000
2022	002766	010000
2023	002770	020000
2024	002772	040000
2025	002774	100000
2026	002776	177776
2027	003000	177775
2028	003002	177773
2029	003004	177767
2030	003006	177757
2031	003010	177737
2032	003012	177677
2033	003014	177577
2034	003016	177377
2035	003020	176777
2036	003022	175777
2037	003024	173777
2038	003026	167777
2039	003030	157777
2040	003032	137777
2041	003034	077777
2042	003036	000000

.WORD	125252
.WORD	052525
.WORD	000000
.WORD	177777
.WORD	000001
.WORD	000002
.WORD	000004
.WORD	000010
.WORD	000020
.WORD	000040
.WORD	000100
.WORD	000200
.WORD	000400
.WORD	001000
.WORD	002000
.WORD	004000
.WORD	010000
.WORD	020000
.WORD	040000
.WORD	100000
.WORD	177776
.WORD	177775
.WORD	177773
.WORD	177767
.WORD	177757
.WORD	177737
.WORD	177677
.WORD	177577
.WORD	177377
.WORD	176777
.WORD	175777
.WORD	173777
.WORD	167777
.WORD	157777
.WORD	137777
.WORD	077777
.WORD	000000

CVDMAA.P11 12-DEC-80 15:59

\*\*\*\*\* DATA PATTERN F \*\*\*\*\*

PATG:

.SBTTL \*\*\*\*\* DATA PATTERN RESULTS TABLE FOR MASTER CLEAR (RESFMC) \*\*\*\*\*

2043 003040  
 2044  
 2045  
 2046  
 2047  
 2048 003040 000020  
 2049 003042 000  
 2050 003043 200  
 2051 003044 000  
 2052 003045 000  
 2053 003046 033  
 2054 003047 000  
 2055 003050 305  
 2056 003051 000  
 2057 003052 000  
 2058 003053 000  
 2059 003054 000  
 2060 003055 000  
 2061 003056 000  
 2062 003057 000  
 2063 003060 000  
 2064 003061 000  
 2065  
 2066  
 2067  
 2068 003062 000020  
 2069

.EVEN  
 RESFMC: .WORD RESFT3-RESFMC-2  
 BSELRS: .BYTE 000 :BSEL0  
 .BYTE 200 :BSEL1 -- 'RUN' BIT SET  
 .BYTE 000 :BSEL2  
 .BYTE 000 :BSEL3  
 .BYTE 033 :BSEL4 -- CODE FOR THE DMV-11  
 .BYTE 000 :BSEL5  
 .BYTE 305 :BSEL6 -- INDICATING VALID COMPLETION OF U-DIAG.  
 .BYTE 000 :BSEL7  
 .BYTE 000 :BSEL10  
 .BYTE 000 :BSEL11  
 .BYTE 000 :BSEL12  
 .BYTE 000 :BSEL13  
 .BYTE 000 :BSEL14  
 .BYTE 000 :BSEL15  
 .BYTE 000 :BSEL16  
 .BYTE 000 :BSEL17

.SBTTL \*\*\*\*\* DATA PATTERN RESULTS FOR TEST 3 (RESFT3) \*\*\*\*\*

RESFT3: .BLKW 16.  
.EVEN



CVDMAA.P11 12-DEC-80 15:59

DATA BUFFER AREAS

2070  
 2071  
 2072 003122 000400  
 2073  
 2074  
 2075  
 2076  
 2077  
 2078 003322  
 2079 003324  
 2080 003326  
 2081 003330  
 2082 003332  
 2083 003334  
 2084 003336  
 2085 003340  
 2086 003342  
 2087 003344  
 2088 003346  
 2089 003350  
 2090 003352  
 2091 003354  
 2092 003356  
 2093 003360  
 2094  
 2095 003122  
 2096 003206

.SBTTL DATA BUFFER AREAS

BUFAREA: .BLKB 256.

: THIS BUFFER HAS SOME ALTERNATE USES TOO. THE FOLLOWING LABELS ARE PROVIDED  
: FOR THOSE USAGES.

W0 = BUFAREA+128.  
 W1 = W0+2  
 W2 = W1+2  
 W3 = W2+2  
 W4 = W3+2  
 W5 = W4+2  
 W6 = W5+2  
 W7 = W6+2  
 W8 = W7+2  
 W9 = W8+2  
 WA = W9+2  
 WB = WA+2  
 WC = WB+2  
 WD = WC+2  
 WE = WD+2  
 WF = WE+2

:THIS WORD TABLE STARTS IN THE MIDDLE OF 'BUFAREA'  
:AND IS USED BY 'ERR6' FOR PRINTING BYTES

BT1 = BUFAREA  
BT2 = BUFAREA+64

:BYTE TABLE # 1  
:BYTE TABLE # 2

CVDMAA.P11 12-DEC-80 15:59

GLOBAL TEXT SECTION

.SBTTL GLOBAL TEXT SECTION

```

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

```

```

:*****
:* NAMES OF DEVICES SUPPORTED BY PROGRAM
:-----
  DEVTYP <M8053 OR M8064>

```

```

LSDVTYP::
          .ASCIZ /M8053 OR M8064/

```

.EVEN

```

:*****
:* TITLE OF PROGRAM
:-----

```

```

.RADIX 10.
  DESCRIPT <DMV-11 U-CONTRL LOGIC DIAG - PART 1 OF 2>

```

```

LSDDESC::
          .ASCIZ /DMV-11 U-CONTRL

```

.EVEN

.RADIX 8.

```

2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108 003522
2109 003522
2110 003522 034115 032460 020063
2111 003530 051117 046440 030070
2112 003536 032066 000
2113
2114
2115
2116
2117
2118
2119 000012
2120 003542
2121 003542
2122 003542 046504 026526 030461
2123 003550 052440 041455 047117
2124 003556 051124 020114 047514
2125 003564 044507 020103 044504
2126 003572 043501 026440 050040
2127 003600 051101 020103 020061
2128 003606 043117 031040 000
2129
2130 003614
2131 000010

```



CVDMAA.P11 12-DEC-80 15:59

## GLOBAL SUBROUTINES

.SBTTL GLOBAL SUBROUTINES

```

://////
:/ THE GLOBAL SUBROUTINES ARE CALLED BY MORE THAN ONE TEST
://////

```

```

:*****
.SBTTL MASCLR - MASTER CLEAR SUBROUTINE

```

FUNCTION:

```

: THIS SUBROUTINE FORCES THE 6502 MICROPROCESSOR TO EXECUTE A MINI 17 PART
: DIAGNOSTIC OF THE MICRO-PROCESSOR INSTRUCTION SET, RAM DATA AND ADDRESSING
: VALIDITY, AND A ROM CRC TEST. THE CLEAR SUBROUTINE EXECUTES IN
: APPROXIMATELY 500 HUNDRED(S) MILLISECOND. THIS SUBROUTINE WILL SEND THE
: MASTER CLEAR COMMAND AND DELAY FOR APPROX. 500 MSEC. AT WHICH POINT IN
: TIME, THE STATE OF THE CSR REGISTERS IS TESTED. IF ANY ONE OF THE
: REGISTERS CONTAINS ANYTHING THAT IS NOT EXPECTED, AN ERROR IS QUEUE UP AND
: THE CARRY BIT IS SET. ELSE, THE CARRY BIT IS CLEARED.

```

CALLING SEQUENCE:

```

: JSR PC,MASCLR
: BCC NS ;IF NO ERROR OCCURED, PROCEED WITH ROUTINE
: ERROR ;AN ERROR MESSAGE HAS BEEN STACKED: PRINT IT
: <ANY OTHER SPECIAL ERROR PROCESSING MAY BE DONE HERE (I.E. CKLOOP)>

```

NS: &lt;RESUMPTION OF NORMAL PROCESSING&gt;

```

:-----*****
MASCLR: MOV R1,-(SP) ; SAVE REGISTER ONE
: MOVB #RUN!MCLR,@BSEL1 ;SET BOTH THE RUN AND MASTER CLEAR BITS
: ;TO INITIATE THE MICRODIAGNOSTIC
: ;NOW DELAY LONG ENOUGH FOR THE MICRODIAGNOSTIC TO COMPLETE
: MOV DELAY1,R1 ;INITIALIZE THE LOOP COUNTER FOR DELAY LOOP
2$: BEQ 1$ ; EXIT DELAY LOOP IF THE TIME HAS EXPIRED
: DEC R1 ; ELSE, DECREMENT THE LOOP COUNTER AND
: BR 2$ ; CONTINUE TO LOOP.
: ; TIME-UP!
1$: BITB #RUN,@BSEL1 ;CHECK THE RUN BIT --
: BEQ 3$ ;IF NOT SET, GO REPORT THE ERROR
: ;IF THE RUN BIT IS SET, MICRODIAGNOSTICS ARE COMPLETE.
: ;CHECK IF ALL MICRODIAGNOSTICS PASSED.
: CMPB @BSEL6,BSELRS+6 ;THIS CHECKS THE BYTE IN B-SELECT 6 FOR THE
2181: 003646 127737 176514 003050 4$: ;VALID MICRODIAGNOSTIC COMPLETION CODE.
2182: ;IF BAD, GO REPORT ERROR
2183: 003654 001004 ;BNE 3$
2184: ;ELSE, CHECK FOR THE VALID CODE FOR A DMV-11
2185: 003656 127737 176500 003046 ;CMPB @BSEL4,BSELRS+4
2186: 003664 001420 ;BEQ 6$
2187: ;IF THIS TOO IS CORRECT THEN NO ERROR EXISTS
: ;ELSE, FALL INTO THE ERROR REPORTING CODE

```

2132  
2133  
2134  
2135  
2136  
2137  
2138  
2139  
2140  
2141  
2142  
2143  
2144  
2145  
2146  
2147  
2148  
2149  
2150  
2151  
2152  
2153  
2154  
2155  
2156  
2157  
2158  
2159  
2160  
2161  
2162  
2163  
2164  
2165  
2166  
2167  
2168  
2169  
2170  
2171  
2172  
2173  
2174  
2175  
2176  
2177  
2178  
2179  
2180  
2181  
2182  
2183  
2184  
2185  
2186  
2187

003614 010146  
003616 112777 000300 176530  
003624 013701 002316  
003630 001402  
003632 005301  
003634 000775  
003636 132777 000200 176510  
003644 001410  
003646 127737 176514 003050  
003654 001004  
003656 127737 176500 003046  
003664 001420

CVDMAA.P11 12-DEC-80 15:59

## MASCLR - MASTER CLEAR SUBROUTINE

```

2188
2189 003666 004737 004434      3$:   JSR   PC,GETBSR      ;GET THE BSEL REGISTERS FOR DUMPING
2190 003672                                ;MASTER CLEAR ERROR
2191                                ;      QUEUE 'DEVICE FATAL' ERROR # 1
2192 003672 012737 000001 002236                                MOV   #T.EDF,ERRTYP
2193 003700 012737 000001 002240                                MOV   #1,ERRNBR
2194 003706 012737 003734 002242                                MOV   #20$,ERRMSG
2195 003714 012737 005414 002244                                MOV   #ERR3,ERRBLK
2196 003722 000261                                SEC
2197 003724 000401                                BR    7$                ;INDICATE TO THE CALLING ROUTINE THAT
2198                                ;      AN FRROR WAS DETECTED
2199 003726 000241                                6$:   CLC
2200 003730 012601                                7$:   MOV   (SP)+,R1      ;CLEAR THE CARRY BIT TO INDICATE NO ERROR
2201 003732 000207                                RTS   PC              ;RESTORE REGISTER ONE
2202                                BEX   PC              ; RETURN TO THE CALLER
2203 003734 040515 052123 051105  .NLIST
                                20$:  .ASCIZ /MASTER CLEAR FAILURE/
                                .LIST
                                .EVEN

```



CVDMAA.P11 12-DEC-80 15:59

M-LOOP -- MSTCLR -- MASTER CLEAR & ENTER M-LOOP

```

.SBTTL M-LOOP -- MSTCLR -- MASTER CLEAR & ENTER M-LOOP
+*****
MSTCLR -- MASTER CLEAR & ENTER M-LOOP
CALLING SEQUENCE:
      JSR      PC,MSTCLR
      BCC     NS           ;IF NO ERROR OCCURED, PROCEED WITH ROUTINE
      ERROR   ;AN ERROR MESSAGE HAS BEEN STACKED: PRINT IT
      <ANY OTHER SPECIAL ERROR PROCESSING MAY BE DONE HERE (I.E. CKLOOP)>
NS:   <RESUMPTION OF NORMAL PROCESSING>
-----*****

```

```

2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219 003762 012777 140400 176362 MSTCLR: MOV      #<RUN!MCLR!MREQ>*256.,@SEL0 ;INITIATE M-LOOP
2220
2221 003770 010346              MOV      R3,-(SP)
2222 003772 012703 000014      MOV      #12.,R3           ;WAIT FOR THE M-LOOP TO FINISH THE OPERATION
2223 003776 077301 1$:      SOB     R3,1$
2224 004000 012603              MOV      (SP)+,R3
2225
2226 004002 132777 000200 176346 BITB     #MRDY,@SEL2      ;DID THE M-LOOP FINISH
2227 004010 001023              BNE     5$              ;YES, GOOD. RETURN
2228 004012 004737 004576      JSR     PC,GETWSR      ;GET BYTE SELECT REGISTERS
2229 004016 012737 000301 002310 MOV      #RUN!MCLR!MREQ,GDATA ;IDENTIFY REQUESTED FUNCTION
2230 004024              GTDF    EM3,ERR4      ;'MRDY' TIMEOUT
2231              ;          QUEUE 'DEVICE FATAL' ERROR # 2
2232 004024 012737 000001 002236              MOV      #T.EDF,ERRTYP
2233 004032 012737 000002 002240              MOV      #2,ERRNBR
2234 004040 012737 014454 002242              MOV      #EM3,ERRMSG
2235 004046 012737 005426 002244              MOV      #ERR4,ERRBLK
2236 004054 000261              SEC
2237 004056 000401              BR      9$           ;SET CARRY TO INDICATE ERROR
2238 004060 000241 5$:      CLC     ;EXIT WITH THE 'ERROR' FLAG (CARRY BIT) SET
2239 004062 000207 9$:      RTS     PC        ;CLEAR C BIT FOR NO ERRORS
                          ;RETURN

```

CVDMAA.P11 12-DEC-80 15:59

M-LOOP -- READ

```

2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257 004064 012577 176272
2258 004070 112777 000001 176260
2259
2260 004076 010346
2261 004100 012703 000032
2262 004104 077301
2263 004106 012603
2264
2265 004110 132777 000200 176240
2266 004116 001023
2267
2268 004120 004737 004576
2269 004124 012737 000001 002310
2270 004132
2271
2272 004132 012737 000001 002236
2273 004140 012737 000003 002240
2274 004146 012737 014500 002242
2275 004154 012737 005426 002244
2276 004162 000261
2277 004164 000401
2278
2279 004166 000241
2280 004170 117735 176172
2281 004174 000205

```

```

.SBTTL M-LOOP -- READ
*****
: READ - READ THE SPECIFIED ADDRESS WITHIN THE DMV-11
:
: CALLING SEQUENCE:
:
: JSR R5,READ
: .WORD <ADDRESS OF REGISTER WITHIN DMV-11>
: .WORD <DESTINATION ADDRESS WITHIN LSI-11>
: BCC NS ;IF NO ERROR OCCURED, PROCEED WITH ROUTINE
: ERROR ;AN ERROR MESSAGE HAS BEEN STACKED: PRINT IT
: <ANY OTHER SPECIAL ERROR PROCESSING MAY BE DONE HERE (I.E. CKLOOP)>
:
: NS: <RESUMPTION OF NORMAL PROCESSING>
:
:-----*****
READ: MOV (R5)+,@SEL4 ;SETUP SOURCE POINTER
      MOVB #REDLOC,@BSEL2 ;TELL M-LOOP TO GIVE US THE REQUESTED DATA
:
: MOV R3,-(SP)
: MOV #26,R3 ;WAIT FOR THE M-LOOP TO FINISH THE OPERATION
1$: SOB R3,1$
   MOV (SP)+,R3
:
: BITB #MRDY,@BSEL2 ;DID THE M-LOOP FINISH
: BNE 5$ ;YES, GOOD. RETURN
:
: JSR PC,GETWSR ;GET BYTE SELECT REGISTERS
: MOV #REDLOC,GDATA ;IDENTIFY REQUESTED FUNCTION
: GTDF EM4,ERR4 ;'MRDY' TIMEOUT
: ; QUEUE 'DEVICE FATAL' ERROR # 3
:
: MOV #T.EDF,ERRTYP
: MOV #3,ERRNBR
: MOV #EM4,ERRMSG
: MOV #ERR4,ERRBLK
:
: SEC ;INDICATE AN ERROR HAS BEEN STACKED
: BR 6$ ;RETURN WITH THAT INDICATION
:
: 5$: CLC ;INDICATE 'NO ERROR'
: 6$: MOVB @BSEL6,@(R5)+ ;PUT DATA WHERE CALLER WANTS IT
: RTS R5 ;RETURN

```





CVDMAA.P11 12-DEC-80 15:59

M-LOOP -- WRITE

2325  
2326  
2327  
2328  
2329  
2330  
2331  
2332  
2333  
2334  
2335  
2336  
2337  
2338  
2339  
2340  
2341  
2342  
2343  
2344

004310 012577 176046  
004314 113577 176046  
004320 000404

```

.SBTTL M-LOOP -- WRITE
+*****
WRITE - WRITE THE SPECIFIED DATA INTO THE SPECIFIED DMV-11 ADDRESS
CALLING SEQUENCE:
      JSR      R5,WRITE
      .WORD   <ADDRESS OF REGISTER WITHIN DMV-11>
      .WORD   <ADDRESS OF DATA BYTE>
      BCC     NS          ;IF NO ERROR OCCURED, PROCEED WITH ROUTINE
      ERROR   NS          ;AN ERROR MESSAGE HAS BEEN STACKED: PRINT IT
      <ANY OTHER SPECIAL ERROR PROCESSING MAY BE DONE HERE (I.E. CKLOOP)>
NS:   <RESUMPTION OF NORMAL PROCESSING>
-----*****
WRITE: MOV      (R5)+,@SEL4      ;SETUP SOURCE POINTER
      MOVB    @<(R5)+,@SEL6     ;MAKE DATA AVAILABLE TO M-LOOP
      BR      MLWRI             ;THE REST OF THIS ROUTINE IS THE SAME AS 'WRITEI'

```



CVDMAA.P11 12-DEC-80 15:59

M-LOOP -- WRITE IMMEDIATE

2345  
2346  
2347  
2348  
2349  
2350  
2351  
2352  
2353  
2354  
2355  
2356  
2357  
2358  
2359  
2360  
2361  
2362  
2363  
2364  
2365  
2366  
2367  
2368  
2369  
2370  
2371  
2372  
2373  
2374  
2375  
2376  
2377  
2378  
2379  
2380  
2381  
2382  
2383  
2384  
2385  
2386

004322  
004322 012577 176034  
004326 012577 176034  
004332 112777 000002 176016  
004340 010346  
004342 012703 000050  
004346 077301  
004350 012603  
004352 132777 000200 175776  
004360 001023  
004362 004737 004576  
004366 012737 000002 002310  
004374  
004374 012737 000001 002236  
004402 012737 000005 002240  
004410 012737 014500 002242  
004416 012737 005426 002244  
004424 000261  
004426 000401  
004430 000241  
004432 000205

.SBTTL M-LOOP -- WRITE IMMEDIATE  
\*\*\*\*\*  
: WRITEI - WRITE IMMEDIATE THE SPECIFIED DATA INTO THE SPECIFIED DMV-11 ADDRESS  
: CALLING SEQUENCE:  
: JSR R5,WRITEI  
: .WORD <ADDRESS OF REGISTER WITHIN DMV-11>  
: .WORD <DATA FIELD -- DATA TO BE WRITTEN IN DMV-11>  
: BCC NS ;IF NO ERROR OCCURED, PROCEED WITH ROUTINE  
: ERROR ;AN ERROR MESSAGE HAS BEEN STACKED: PRINT IT  
: <ANY OTHER SPECIAL ERROR PROCESSING MAY BE DONE HERE (I.E. CKLOOP)>  
: NS: <RESUMPTION OF NORMAL PROCESSING>  
:-----\*\*\*\*\*

WRITEI:  
MOV (R5)+,@SEL4 ;SETUP SOURCE POINTER  
MOV (R5)+,@SEL6 ;MAKE DATA AVAILABLE TO M-LOOP  
MLWRI: MOVB #WRILOC,@BSEL2 ;TELL M-LOOP TO WRITE THE DATA  
MOV R3,-(SP)  
1\$: MOV #40,R3 ;WAIT FOR THE M-LOOP TO FINISH THE OPERATION  
SOB R3,1\$  
MOV (SP)+,R3  
BITB #MRDY,@BSEL2 ;DID THE M-LOOP FINISH  
BNE 5\$ ;YES, GOOD. RETURN  
JSR PC,GETWSR ;GET BYTE SELECT REGISTERS  
MOV #WRILOC,GDATA ;IDENTIFY REQUESTED FUNCTION  
GDF EM4,ERR4 ;'MRDY' TIMEOUT  
: QUEUE 'DEVICE FATAL' ERROR # 5  
MOV #T.EDF,ERRTYP  
MOV #5,ERRNBR  
MOV #EM4,ERRMSG  
MOV #ERR4,ERRBLK  
SEC ;INDICATE AN ERROR HAS BEEN STACKED  
BR 6\$ ;RETURN WITH THAT INDICATION  
5\$: CLC ;INDICATE 'NO ERROR'  
6\$: RTS R5 ;RETURN

CVDMAA.P11 12-DEC-80 15:59

GETBSR -- GET BYTE SELECT REGISTERS

.SBTTL GETBSR -- GET BYTE SELECT REGISTERS

```

:*****
:
: GET THE CONTENTS OF ALL CONTROL AND STATUS REGISTERS
:
: FUNCTION - THIS SUBROUTINE COLLECTS THE CONTENTS OF THE
:           BYTE SELECT REGISTERS FOR THE PURPOSE OF DISPLAY.
:
: ENTRY CONDITIONS - NONE      ## #   ### #   ## #
:                               # # #   #   ## #   ## #
: EXIT CONDITIONS - NONE      # # #   ## #   #   ##
:                               # # #   #   ##### #   #
:
: REGISTERS DESTROYED - NONE  ##   ###   ### #   #   #
:*****

```

2387  
2388  
2389  
2390  
2391  
2392  
2393  
2394  
2395  
2396  
2397  
2398  
2399  
2400  
2401  
2402  
2403  
2404  
2405  
2406  
2407  
2408  
2409  
2410  
2411  
2412  
2413  
2414  
2415  
2416  
2417  
2418  
2419  
2420  
2421  
2422  
2423  
2424  
2425  
2426  
2427  
2428  
2429  
2430  
2431  
2432  
2433

004434	117737	175712	002246
004442	117737	175706	002250
004450	117737	175702	002252
004456	117737	175676	002254
004464	117737	175672	002256
004472	117737	175666	002260
004500	117737	175662	002262
004506	117737	175656	002264
004514	117737	175652	002266
004522	117737	175646	002270
004530	117737	175642	002272
004536	117737	175636	002274
004544	117737	175632	002276
004552	117737	175626	002300
004560	117737	175622	002302
004566	117737	175616	002304
004574	000207		
004576	017737	175550	002246
004604	017737	175546	002250
004612	017737	175544	002252
004620	017737	175542	002254
004626	017737	175540	002256
004634	017737	175536	002260
004642	017737	175534	002262
004650	017737	175532	002264
004656	000207		

```

GETBSR:  MOV  @BSSEL0,BSR0      ;PUT THE CURRENT CSR VALUES INTO THE PRINT-OUT
          MOV  @BSSEL1,BSR1      ;TABLE
          MOV  @BSSEL2,BSR2
          MOV  @BSSEL3,BSR3
          MOV  @BSSEL4,BSR4
          MOV  @BSSEL5,BSR5
          MOV  @BSSEL6,BSR6
          MOV  @BSSEL7,BSR7
          MOV  @BSSEL10,BSR10
          MOV  @BSSEL11,BSR11
          MOV  @BSSEL12,BSR12
          MOV  @BSSEL13,BSR13
          MOV  @BSSEL14,BSR14
          MOV  @BSSEL15,BSR15
          MOV  @BSSEL16,BSR16
          MOV  @BSSEL17,BSR17
          RTS      PC      ;RETURN TO CALLER

```

.SBTTL GETWSR -- GET WORD SELECT REGISTERS  
; 'WORD' VERSION OF ABOVE SUBROUTINE

```

GETWSR:  MOV  @ASEL0,WSR0      ;MOVE THE 8 WORD REGISTERS TO THE OTHERWISE
          MOV  @ASEL2,WSR2      ;BYTE TABLE
          MOV  @ASEL4,WSR4
          MOV  @ASEL6,WSR6
          MOV  @ASEL10,WSR10
          MOV  @ASEL12,WSR12
          MOV  @ASEL14,WSR14
          MOV  @ASEL16,WSR16
          RTS      PC      ;RETURN TO CALLER

```



CVDMAA.P11 12-DEC-80 15:59

.INITT1 -- INITIALIZE TIMER # 1

.SBTTL .INITT1 -- INITIALIZE TIMER # 1

2434  
2435  
2436  
2437  
2438  
2439  
2440  
2441  
2442  
2443  
2444  
2445  
2446  
2447  
2448  
2449  
2450  
2451  
2452  
2453  
2454  
2455  
2456  
2457  
2458  
2459  
2460  
2461  
2462  
2463  
2464  
2465  
2466  
2467  
2468  
2469  
2470  
2471  
2472  
2473  
2474  
2475  
2476  
2477  
2478  
2479  
2480  
2481  
2482  
2483  
2484  
2485  
2486  
2487  
2488  
2489

004660 010146  
004662 112537 002455  
004666 112537 002457  
004672 111537 002467  
004676 142737 177477 002467  
004704 012501  
  
004706 106301  
004710 042701 177677  
004714 140177 175440  
004720 106301  
004722 052701 000100  
004726 110137 002475  
  
004732 004537 004310  
004736 120016  
004740 002475  
004742 103431  
  
004744 004537 004064  
004750 120013  
004752 002466

```

*****
* INITT1 - INITIALIZE TIMER # 1
*
*   CALLING SEQUENCE:
*
*       JSR     R5,INITT1
*       .WORD  <VALUE LOADED INTO THE T1 LATCH @ T1LL & T1LH>
*       .WORD  <BITS 6 & 7 WILL BE LOADED INTO 'ACR', BIT 5 WILL BE
*                USED TO SET OR CLEAR BIT 6 ('T1') OF THE INTERRUPT
*                ENABLE REGISTER ('IER')>
*
*   SEQUENCE OF EVENTS HEREIN:
*
*       SET THE VIA'S INTERRUPT ENABLE REGISTER ('IER')
*
*       SET THE VIA'S 'ACR'
*
*       SET T1L-L (ADDR 06)
*
*       SET T1L-H (ADDR 07)
*
*       RETURN WITHOUT ANY ERROR CHECKING
*****

```

```

INITT1: MOV     R1,-(SP)           ;SAVE THE REGISTER WE WILL BE USING
        MOVB   (R5)+,TMP6+1     ;SETUP VALUES TO BE LOADED INTO THE LATCHES
        MOVB   (R5)+,TMP7+1
        MOVB   (R5),TMPB+1      ;GET & PROCESS BITS FOR ACR 6 & 7
        BICB   #^C<BIT6+BIT7>,TMPB+1 ;EXTRACT BITS 6 & 7 & SAVE THEM FOR LATER
        MOV    (R5)+,R1         ;NOW, GET THE BIT TO BE USED IN SETTING OR
                                ;CLEARING BIT 6 OF 'IER'

; THE PASSED BIT IS IN THE WRONG POSITION BUT, IT SHOULD CONTROL THE OPERATION.
; WE KNOW WE ARE SETTING OR CLEARING BIT 6 -- THUS, THE PASSED BIT WILL BECOME
; THE CONTROLLING BIT 7 AND WE WILL 'OR' IN THE BIT WE WISH TO BE CONTROLLED
; (BIT 6).

        ASLB   R1               ;THIS PUTS THE PASSED BIT INTO BIT 6.
        BIC   #^C<BIT6>,R1     ;WHILE HERE, CLEAR ALL OTHER BITS AND
        BICB  R1,@SEL3        ;CLEAR THE INTERRUPT FLAG IN THE SELECT REG.
        ASLB  R1               ;NOW THE BIT IS IN THE CONTROLLING POSITION
        BIS   #BIT6,R1        ;SET BIT 6
        MOVB  R1,TMPE+1       ;THE CALL WILL NOW WRITE THE APPROPRIATE VALUE

        JSR   R5,WRITE        ;WRITE TO
                                ;THE VIA'S IER
                                ;INTERRUPT ENABLE/DISABLE INFORMATION
        TMBE+1
        BCS   638             ;EXIT ON ERROR

        JSR   R5,READ         ;READ THE CURRENT SETTING OF
                                ;THE VIA'S ACR
        ACR
        TMPE

```

CVDMAA.P11 12-DEC-80 15:59

.INITT1 -- INITIALIZE TIMER # 1

```

2490 004754 103424          BCS      63$          ;EXIT ON ERROR
2491
2492 004756 013701 002466  MOV      TMPB,R1      ;GET THAT VALUE
2493 004762 042701 177477  BIC      #^C<BIT6+BIT7>,R1 ;CLEAR BITS 6 & 7
2494 004766 150137 002467  BISB    R1,TMPB+1     ;ADD CURRENT BITS 0 --> 5 TO NEW BITS 6 & 7
2495
2496 004772 004537 004310  JSR      R5,WRITE     ;WRITE THE NEW REGISTER SETTING TO VIA'S ACR
2497 004776 120013
2498 005000 002467
2499 005002 103411          BCS      63$          ;EXIT ON ERROR
2500
2501 005004 004537 004310  JSR      R5,WRITE     ;WRITE TO
2502 005010 120006          T1LL                     ;LOW ORDER LATCH REGISTER (T1L-L)
2503 005012 002455          TMP6+1                   ;THE VALUE PASSED
2504 005014 103404          BCS      63$          ;EXIT ON ERROR
2505
2506 005016 004537 004310  JSR      R5,WRITE     ;WRITE TO
2507 005022 120007          T1LH                     ;HIGH ORDER LATCH REGISTER (T1L-H)
2508 005024 002457          TMP7+1                   ;THE VALUE PASSED
2509
2510
2511 005026 012601          63$:  MOV      (SP)+,R1   ;RESTORE R1
2512 005030 000205          RTS      R5            ;RETURN
2513
2514
2515
2516
2517
2518
2519
2520 005032 000207
2521

```

```

.SBTTL STALL -- DELAY FOR 10.5 MICRO-SEC'S (ON LSI-11)
;*****
; STALL -- THIS SUBROUTINE STALLS FOR ABOUT 10.5 MICRO-SECONDS
;-----

```

STALL: RTS PC



CVDMAA.P11 12-DEC-80 15:59

STREG -- STATIC TEST OF SPECIFIED DMV-11 LOCATION

.SBTTL STREG -- STATIC TEST OF SPECIFIED DMV-11 LOCATION

\*\*\*\*\*  
STREG -- PERFORM A STATIC TEST OF THE SPECIFIED REGISTER

CALLING SEQUENCE:

<R0 CONTAINS THE ADDRESS OF THE REGISTER TO BE TESTED>  
<'TDATA' CONTAINS THE TEST BYTE>  
<'GDATA' CONTAINS THE EXPECTED DATA>

JSR PC,STREG  
BCC NS ;IF NO ERROR OCCURED, PROCEED WITH ROUTINE  
ERROR ;AN ERROR MESSAGE HAS BEEN STACKED: PRINT IT  
<ANY OTHER SPECIAL ERROR PROCESSING MAY BE DONE HERE (I.E. CKLOOP)>

NS: <RESUMPTION OF NORMAL PROCESSING>

\*\*\*\*\*

2522  
2523  
2524  
2525  
2526  
2527  
2528  
2529  
2530  
2531  
2532  
2533  
2534  
2535  
2536  
2537  
2538  
2539  
2540  
2541  
2542  
2543  
2544  
2545  
2546  
2547  
2548  
2549  
2550  
2551  
2552  
2553  
2554  
2555  
2556  
2557  
2558  
2559  
2560  
2561  
2562  
2563  
2564  
2565  
2566  
2567  
2568

005034 010037 005050  
005040 010037 005062  
005044 004537 004310  
005050 000000  
005052 002306  
005054 103435  
005056 004537 004064  
005062 000000  
005064 002312  
005066 103430  
005070 123737 002310 002312  
005076 000241  
005100 001423  
005102 013737 005050 002334  
005110 042737 177760 002334  
005116  
005116 012737 000001 002236  
005124 012737 000006 002240  
005132 012737 015565 002242  
005140 012737 006612 002244  
005146 000261  
005150 000207

STREG: MOV R0,2\$ ;PUT SPECIFIED REGISTER'S ADDRESS IN I/O CALLS  
MOV R0,4\$  
  
2\$: JSR R5,WRITE ;WRITE IT  
0 ;\*\*\* MODIFIED FROM ABOVE \*\*\*  
TDATA ;\*\*\* MODIFIED FROM ABOVE \*\*\*  
BCS 10\$ ;ON ERROR, EXIT  
  
4\$: JSR R5,READ ;READ IT BACK AGAIN  
0 ;\*\*\* MODIFIED FROM ABOVE \*\*\*  
BDATA  
BCS 10\$ ;ON ERROR, EXIT  
  
CMPB GDATA,BDATA ;DID WE READ WHAT WE WROTE?  
CLC ; (THIS ISN'T NEEDED FOR THE ERROR TEST BUT  
; MUST BE CLEARED ON EXIT IF NO ERROR OCCURED)  
BEQ 10\$ ;YES, EXIT FROM SUBTEST  
MOV 2\$,REGNUM ;BUILD REGISTER #  
BIC #177760,REGNUM  
GTDF EM25,ERR7 ;REPORT READ/WRITE ERROR  
; QUEUE 'DEVICE FATAL' ERROR # 6  
MOV #T.EDF,ERRTYP  
MOV #6,ERRNBR  
MOV #EM25,ERRMSG  
MOV #ERR7,ERRBLK  
  
10\$: SEC ;INDICATE THAT AN ERROR WAS DETECTED  
RTS PC

CVDMAA.P11 12-DEC-80 15:59

INTERRUPT HANDLER -- MPIHAN

.SBTTL INTERRUPT HANDLER -- MPIHAN

```

:*****
:MPIHAN -- COUNT INTERRUPTS -- USUALLY INTERRUPT 'A'
:
:   THIS ROUTINE WILL INCREMENT THE LOW BYTE OF 'INTFLG' EACH TIME IT IS
:   ENTERED.  IF 'IHILNK' IS NON-ZERO, VECTOR TO THE ADDRESS THEREIN USING
:   A 'JSR PC'
:*****

```

```

2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579 005152
2580 005152
2581 005152 010046
2582 005154 105737 002330
2583 005160 001007
2584 005162 004737 004434
2585 005166
2586
2587 005166 104455
2588 005170 000007
2589 005172 015613
2590 005174 005414
2591 005176 000407
2592
2593 005200 105237 002326
2594 005204 005737 005222
2595 005210 001402
2596 005212 004777 000004
2597 005216 012600
2598 005220
2599 005220
2600 005220 000002
2601
2602 005222 000000

```

```

BGNSRV MPIHAN
MOV R0,-(SP) ;SAVE R0
TSTB INTWCH ;HAVE WE BEEN TOLD TO WATCH FOR TYPE 'A' INT'S?
BNE 5$ ;YES, DO NORMAL INTERRUPT PROCESSING
JSR PC,GETBSR ;NO, DUMP REGISTERS AND
GEDF EM34,ERR3 ; REPORT 'UNEXPECTED INTERRUPT'
; 'DEVICE FATAL' ERROR # 7
TRAP CSERDF
.WORD 7
.WORD EM34
.WORD ERR3

BR 10$ ;GO TO EXIT

5$: INCB INTFLG ;INCREMENT LOW BYTE OF INTERRUPT COUNTER
TST IHILNK ;ARE WE EXPECTED TO EXECUTE ANOTHER ROUTINE?
BEQ 10$ ;NO, GET OUT
JSR PC,@IHILNK ;YES, GO TO IT -- I HOPE IT'S VALID!
10$: MOV (SP)+,R0 ;RESTORE R0
ENDSRV ;RETURN TO INTERRUPTED PROCESS
L10002:
RTI

IHILNK: .WORD 0 ;POINTER TO AUXILIARY INT. HANDLING ROUTINE

```



CVDMAA.P11 12-DEC-80 15:59

INTERRUPT HANDLER -- MPOHAN

.SBTTL INTERRUPT HANDLER -- MPOHAN

```

:*****
:MPOHAN -- SIMPLY COUNT INTERRUPTS -- USUALLY INTERRUPT 'B'
:
:   THIS ROUTINE WILL INCREMENT THE HIGH BYTE OF 'INTFLG' EACH TIME IT IS
:   ENTERED.  IF 'IHOLNK' IS NON-ZERO, VECTOR TO THE ADDRESS THEREIN USING
:   A 'JSR PC'
:*****

```

```

2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613 005224
2614 005224
2615 005224 010046
2616 005226 105737 002331
2617 005232 001007
2618 005234 004737 004434
2619 005240
2620
2621 005240 104455
2622 005242 000010
2623 005244 015644
2624 005246 005414
2625 005250 000407
2626
2627 005252 105237 002327
2628 005256 005737 005274
2629 005262 001402
2630 005264 004777 000004
2631 005270 012600
2632 005272
2633 005272
2634 005272 000002
2635
2636 005274 000000

```

```

BGNSRV MPOHAN
MPOHAN::
MOV R0,-(SP) ;SAVE R0
TSTB INTWCH+1 ;HAVE WE BEEN TOLD TO WATCH FOR TYPE 'B' INT'S?
BNE 5$ ;YES, DO NORMAL INTERRUPT PROCESSING
JSR PC,GETBSR ;NO, DUMP REGISTERS AND
GEDF EM34B,ERR3 ; REPORT 'UNEXPECTED INTERRUPT'
; 'DEVICE FATAL' ERROR # 8
TRAP CSERDF
.WORD 8
.WORD EM34B
.WORD ERR3
BR 10$ ;GO TO EXIT
5$: INCB INTFLG+1 ;INCREMENT HIGH BYTE OF INTERRUPT COUNTER
TST IHOLNK ;ARE WE EXPECTED TO EXECUTE ANOTHER ROUTINE?
BEQ 10$ ;NO, GET OUT
JSR PC,@IHOLNK ;YES, GO TO IT -- I HOPE IT'S VALID!
10$: MOV (SP)+,R0 ;RESTORE R0
ENDSRV ;RETURN TO INTERRUPTED PROCESS
L10003:
RTI
IHOLNK: .WORD 0 ;POINTER TO AUXILIARY INT. HANDLING ROUTINE

```

CVDMAA.P11 12-DEC-80 15:59

GLOBAL ERROR REPORT REPORT SECTION

.SBTTL GLOBAL ERROR REPORT REPORT SECTION

:/ THE GLOBAL ERROR REPORT SECTION CONTAINS ERROR MESSAGES THAT ARE USED IN MORE THAN ONE TEST. /

.SBTTL ERROR HANDLER -- ERR1 -- 'NO NOTHING' HANDLER

BGNMSG ERR1
JSR PC,NULERR ;USE COMMON ROUTINE TO TERMINATE ERROR MESSAGE
ENDMSG
L10004: TRAP CSMSG

.SBTTL ERROR HANDLER -- ERR2 -- CSR REGISTER ERROR REPORTING

BGNMSG ERR2
PRINTB #FMT02,#TXT5,REGNUM
JSR PC,XORGB
PRINTB #FMT02A,<B,GDATA>,<B,BDATA>,<B,XDATA>
ERR2::
MOV REGNUM,-(SP)
MOV #TXT5,-(SP)
MOV #FMT02,-(SP)
MOV #3,-(SP)
MOV SP,R0
TRAP CSPNTB
ADD #10,SP
CLR -(SP)
BISB XDATA,(SP)
CLR -(SP)
BISB BDATA,(SP)
CLR -(SP)
BISB GDATA,(SP)
MOV #FMT02A,-(SP)
MOV #4,-(SP)
MOV SP,R0
TRAP CSPNTB
ADD #12,SP
JSR PC,ERR4\$ ;DUMP THE BYTE SELECT REGISTERS
JSR PC,NULERR ;USE COMMON ROUTINE TO TERMINATE ERROR MESSAGE
ENDMSG
L10005: TRAP CSMSG

.SBTTL ERROR HANDLER -- ERR3 -- DUMP THE BYTE SELECT REGISTERS

BGNMSG ERR3
JSR PC,ERR4\$
JSR PC,NULERR ;USE COMMON ROUTINE TO TERMINATE ERROR MESSAGE
ENDMSG
L10006:

2637
2638
2639
2640
2641
2642
2643
2644
2645
2646
2647 005276
2648 005276
2649 005276 004737 012072
2650 005302
2651 005302
2652 005302 104423
2653
2654
2655
2656 005304
2657 005304
2658 005304
2659 005304 013746 002334
2660 005310 012746 013605
2661 005314 012746 012124
2662 005320 012746 000003
2663 005324 010600
2664 005326 104414
2665 005330 062706 000010
2666 005334 004737 011276
2667 005340
2668 005340 005046
2669 005342 153716 002314
2670 005346 005046
2671 005350 153716 002312
2672 005354 005046
2673 005356 153716 002310
2674 005362 012746 012161
2675 005366 012746 000004
2676 005372 010600
2677 005374 104414
2678 005376 062706 000012
2679 005402 004737 011322
2680 005406 004737 012072
2681 005412
2682 005412
2683 005412 104423
2684
2685
2686
2687 005414
2688 005414
2689 005414 004737 011322
2690 005420 004737 012072
2691 005424
2692 005424



CVDMAA.P11 12-DEC-80 15:59

ERROR HANDLER -- ERR3 -- DUMP THE BYTE SELECT REGISTERS

```

2693 005424 104423
2694
2695
2696
2697 005426
2698 005426
2699 005426 010146
2700 005430 113701 002310
2701 005434 122701 000017
2702 005440 103013
2703 005442
2704 005442 005046
2705 005444 150116
2706 005446 012746 012400
2707 005452 012746 000002
2708 005456 010600
2709 005460 104415
2710 005462 062706 000006
2711 005466 000425
2712
2713 005470 001001
2714 005472 005001
2715 005474 022701 000007
2716 005500 003002
2717 005502 012701 000006
2718 005506 006301
2719 005510
2720 005510 016146 017532
2721 005514 005046
2722 005516 153716 002310
2723 005522 012746 012443
2724 005526 012746 000003
2725 005532 010600
2726 005534 104415
2727 005536 062706 000010
2728
2729 005542 012601
2730 005544 004737 011710
2731 005550
2732 005550
2733 005550 104423
2734
2735
2736
2737 005552
2738 005552
2739 005552
2740 005552 013746 002334
2741 005556 012746 013605
2742 005562 012746 012124
2743 005566 012746 000003
2744 005572 010600
2745 005574 104414
2746 005576 062706 000010
2747 005602 004737 011276
2748 005606
    
```

```

-----
:SBTTL ERROR HANDLER -- ERR4 -- M-LOOP TIMEOUT ERROR HANDLING
-----
      BGNMSG  ERR4
      MOV     R1, -(SP)           ;SAVE THE WORKING REGISTER
      MOVB   GDATA, R1          ;SAVE THIS FOR LATER
      CMPB   #17, R1            ;WAS THIS AN M-LOOP REQUEST?
      BHIS   5$                 ;YES, THEN REPORT THE FUNCTION CODE
      PRINTX #FMT5, <B, R1>     ;NO, THEN IT MUST BE A BSEL1 SETTING
                                  CLR     -(SP)
                                  BISB   R1, (SP)
                                  MOV    #FMT5, -(SP)
                                  MOV    #2, -(SP)
                                  MOV    SP, R0
                                  TRAP   C$PNTX
                                  ADD    #6, SP
      BR     20$

5$:   BNE     6$                 ;IF IT WAS A 17, THIS IS A 'NOP' AND
      CLR     R1                 ; THE TEXT POINTER MUST SO REFLECT.
6$:   CMP     #7, R1            ;IS FUNCTION CODE > 7?
      BGT     7$                 ;NO, THEN WE CAN HANDLE IT
      MOV     #6, R1            ;YES, THEN IT'S UNDEFINED -- SAY SO
7$:   ASL     R1                 ;CONVERT TO A WORD OFFSET
      PRINTX #FMT5A, <B, GDATA>, TXTMLT(R1) ;REPORT THE FAILING FUNCTION
                                  MOV    TXTMLT(R1), -(SP)
                                  CLR    -(SP)
                                  BISB   GDATA, (SP)
                                  MOV    #FMT5A, -(SP)
                                  MOV    #3, -(SP)
                                  MOV    SP, R0
                                  TRAP   C$PNTX
                                  ADD    #10, SP

20$:  MOV     (SP)+, R1          ;RESTORE THE WORKING REGISTER
      JSR    PC, ERR5$         ;DUMP THE SELECT REGISTERS
      ENDMSG

                                  L10007:
                                  TRAP   C$MSG
-----
:SBTTL ERROR HANDLER -- ERR5 -- WORD SELECT REG. ERRORS
-----
      BGNMSG  ERR5
      PRINTB  #FMT02, #TXT5, REGNUM
                                  ERR5::
                                  MOV    REGNUM, -(SP)
                                  MOV    #TXT5, -(SP)
                                  MOV    #FMT02, -(SP)
                                  MOV    #3, -(SP)
                                  MOV    SP, R0
                                  TRAP   C$PNTB
                                  ADD    #10, SP

      JSR    PC, XORGB
      PRINTB #FMT10, GDATA, BDATA, XDATA
    
```

CVDMAA.P11 12-DEC-80 15:59

ERROR HANDLER -- ERR5 -- WORD SELECT REG. ERRORS

2749	005606	013746	002314							
2750	005612	013746	002312					MOV	XDATA,-(SP)	
2751	005616	013746	002310					MOV	BDATA,-(SP)	
2752	005622	012746	012654					MOV	GDATA,-(SP)	
2753	005626	012746	000004					MOV	#FMT10,-(SP)	
2754	005632	010600						MOV	#4,-(SP)	
2755	005634	104414						MOV	SP,R0	
2756	005636	062706	000012					TRAP	C\$PNTB	
2757	005642	004737	011710					ADD	#12,SP	
2758	005646			JSR	PC,ERR5\$				;DUMP THE SELECT REGISTERS	
2759	005646			ENDMSG						
2760	005646	104423							L10010: TRAP C\$MSG	
2761				-----						
2762				.SBTTL ERROR HANDLER -- ERR6 -- VIA REGISTER ERRORS W/FULL REG. DUMP						
2763				-----						
2764	005650			BGNMSG	ERR6					
2765	005650								ERR6::	
2766				;*** PRINT THE FIRST HALF OF THE REGISTERS ***						
2767	005650	010146		MOV	R1,-(SP)				;PRESERVE R1'S CONTENTS	
2768	005652	012701	002604	MOV	#PATCR,R1				;POINT TO EXPECTED VALUES	
2769	005656			PRINTX	#FMT06,#TXT7					
2770	005656	012746	013632						MOV #TXT7,-(SP)	
2771	005662	012746	012562						MOV #FMT06,-(SP)	
2772	005666	012746	000002						MOV #2,-(SP)	
2773	005672	010600							MOV SP,R0	
2774	005674	104415							TRAP C\$PNTX	
2775	005676	062706	000006						ADD #6,SP	
2776	005702			PRINTX	#FMT06A,#TXT8A,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>					
2777	005702	005046							CLR -(SP)	
2778	005704	152116							BISB (R1)+,(SP)	
2779	005706	005046							CLR -(SP)	
2780	005710	152116							BISB (R1)+,(SP)	
2781	005712	005046							CLR -(SP)	
2782	005714	152116							BISB (R1)+,(SP)	
2783	005716	005046							CLR -(SP)	
2784	005720	152116							BISB (R1)+,(SP)	
2785	005722	005046							CLR -(SP)	
2786	005724	152116							BISB (R1)+,(SP)	
2787	005726	005046							CLR -(SP)	
2788	005730	152116							BISB (R1)+,(SP)	
2789	005732	012746	014004						MOV #TXT8A,-(SP)	
2790	005736	012746	012571						MOV #FMT06A,-(SP)	
2791	005742	012746	000010						MOV #10,-(SP)	
2792	005746	010600							MOV SP,R0	
2793	005750	104415							TRAP C\$PNTX	
2794	005752	062706	000022						ADD #22,SP	
2795	005756			PRINTX	#FMT06B,<B,(R1)+>,<B,(R1)+>					
2796	005756	005046							CLR -(SP)	
2797	005760	152116							BISB (R1)+,(SP)	
2798	005762	005046							CLR -(SP)	
2799	005764	152116							BISB (R1)+,(SP)	
2800	005766	012746	012637						MOV #FMT06B,-(SP)	
2801	005772	012746	000003						MOV #3,-(SP)	
2802	005776	010600							MOV SP,R0	
2803	006000	104415							TRAP C\$PNTX	
2804	006002	062706	000010						ADD #10,SP	



CVDMAA.P11 12-DEC-80 15:59

ERROR HANDLER -- ERR6 -- VIA REGISTER ERRORS W/FULL REG. DUMP

2805	006006	012701	003122	MOV	#BT1,R1	:POINT TO ACTUAL VALUES		
2806	006012			PRINTX	#FMT06A,#TXT8B,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>			
2807	006012	005046					CLR	-(SP)
2808	006014	152116					BISB	(R1)+,(SP)
2809	006016	005046					CLR	-(SP)
2810	006020	152116					BISB	(R1)+,(SP)
2811	006022	005046					CLR	-(SP)
2812	006024	152116					BISB	(R1)+,(SP)
2813	006026	005046					CLR	-(SP)
2814	006030	152116					BISB	(R1)+,(SP)
2815	006032	005046					CLR	-(SP)
2816	006034	152116					BISB	(R1)+,(SP)
2817	006036	005046					CLR	-(SP)
2818	006040	152116					BISB	(R1)+,(SP)
2819	006042	012746	014021	MOV	#TXT8B,-(SP)			
2820	006046	012746	012571	MOV	#FMT06A,-(SP)			
2821	006052	012746	000010	MOV	#10,-(SP)			
2822	006056	010600		MOV	SP,R0			
2823	006060	104415		TRAP	CSPNTX			
2824	006062	062706	000022	ADD	#22,SP			
2825	006066			PRINTX	#FMT06B,<B,(R1)+>,<B,(R1)+>			
2826	006066	005046					CLR	-(SP)
2827	006070	152116					BISB	(R1)+,(SP)
2828	006072	005046					CLR	-(SP)
2829	006074	152116					BISB	(R1)+,(SP)
2830	006076	012746	012637	MOV	#FMT06B,-(SP)			
2831	006102	012746	000003	MOV	#3,-(SP)			
2832	006106	010600		MOV	SP,R0			
2833	006110	104415		TRAP	CSPNTX			
2834	006112	062706	000010	ADD	#10,SP			
2835	006116	012701	003206	MOV	#BT2,R1	:POINT TO XOR VALUES		
2836	006122			PRINTX	#FMT06A,#TXT8C,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>			
2837	006122	005046					CLR	-(SP)
2838	006124	152116					BISB	(R1)+,(SP)
2839	006126	005046					CLR	-(SP)
2840	006130	152116					BISB	(R1)+,(SP)
2841	006132	005046					CLR	-(SP)
2842	006134	152116					BISB	(R1)+,(SP)
2843	006136	005046					CLR	-(SP)
2844	006140	152116					BISB	(R1)+,(SP)
2845	006142	005046					CLR	-(SP)
2846	006144	152116					BISB	(R1)+,(SP)
2847	006146	005046					CLR	-(SP)
2848	006150	152116					BISB	(R1)+,(SP)
2849	006152	012746	014036	MOV	#TXT8C,-(SP)			
2850	006156	012746	012571	MOV	#FMT06A,-(SP)			
2851	006162	012746	000010	MOV	#10,-(SP)			
2852	006166	010600		MOV	SP,R0			
2853	006170	104415		TRAP	CSPNTX			
2854	006172	062706	000022	ADD	#22,SP			
2855	006176			PRINTX	#FMT06B,<B,(R1)+>,<B,(R1)+>			
2856	006176	005046					CLR	-(SP)
2857	006200	152116					BISB	(R1)+,(SP)
2858	006202	005046					CLR	-(SP)
2859	006204	152116					BISB	(R1)+,(SP)
2860	006206	012746	012637	MOV	#FMT06B,-(SP)			

CVDMAA.P11 12-DEC-80 15:59

ERROR HANDLER -- ERR6 -- VIA REGISTER ERRORS W/FULL REG. DUMP

```

2861 006212 012746 000003
2862 006216 010600
2863 006220 104415
2864 006222 062706 000010
2865
2866 006226 012701 002614
2867 006232
2868 006232 012746 013717
2869 006236 012746 012562
2870 006242 012746 000002
2871 006246 010600
2872 006250 104415
2873 006252 062706 000006
2874 006256
2875 006256 005046
2876 006260 152116
2877 006262 005046
2878 006264 152116
2879 006266 005046
2880 006270 152116
2881 006272 005046
2882 006274 152116
2883 006276 005046
2884 006300 152116
2885 006302 005046
2886 006304 152116
2887 006306 012746 014004
2888 006312 012746 012571
2889 006316 012746 000010
2890 006322 010600
2891 006324 104415
2892 006326 062706 000022
2893 006332
2894 006332 005046
2895 006334 152116
2896 006336 005046
2897 006340 152116
2898 006342 012746 012637
2899 006346 012746 000003
2900 006352 010600
2901 006354 104415
2902 006356 062706 000010
2903 006362 012701 003132
2904 006366
2905 006366 005046
2906 006370 152116
2907 006372 005046
2908 006374 152116
2909 006376 005046
2910 006400 152116
2911 006402 005046
2912 006404 152116
2913 006406 005046
2914 006410 152116
2915 006412 005046
2916 006414 152116
    
```

\*\*\* PRINT SECOND HALF OF THE REGISTERS \*\*\*

```

MOV #PATCR+8.,R1 ;POINT TO 2ND HALF OF REGISTERS EXPECTED VALUES
PRINTX #FMT06,#TXT7A
    
```

```

MOV #3,-(SP)
MOV SP,R0
TRAP C$PNTX
ADD #10,SP
    
```

```

MOV #TXT7A,-(SP)
MOV #FMT06,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C$PNTX
ADD #6,SP
    
```

```

PRINTX #FMT06A,#TXT8A,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>
    
```

```

CLR -(SP)
BISB (R1)+,(SP)
CLR -(SP)
BISB (R1)+,(SP)
CLR -(SP)
BISB (R1)+,(SP)
CLR -(SP)
BISB (R1)+,(SP)
CLR -(SP)
BISB (R1)+,(SP)
CLR -(SP)
BISB (R1)+,(SP)
CLR -(SP)
BISB (R1)+,(SP)
    
```

```

MOV #TXT8A,-(SP)
MOV #FMT06A,-(SP)
MOV #10,-(SP)
MOV SP,R0
TRAP C$PNTX
ADD #22,SP
    
```

```

PRINTX #FMT06B,<B,(R1)+>,<B,(R1)+>
    
```

```

CLR -(SP)
BISB (R1)+,(SP)
CLR -(SP)
BISB (R1)+,(SP)
MOV #FMT06B,-(SP)
MOV #3,-(SP)
MOV SP,R0
TRAP C$PNTX
ADD #10,SP
    
```

```

MOV #BT1+8.,R1 ;POINT TO 2ND HALF OF ACTUAL VALUES
PRINTX #FMT06A,#TXT8B,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>
    
```

```

CLR -(SP)
BISB (R1)+,(SP)
CLR -(SP)
BISB (R1)+,(SP)
CLR -(SP)
BISB (R1)+,(SP)
CLR -(SP)
BISB (R1)+,(SP)
CLR -(SP)
BISB (R1)+,(SP)
CLR -(SP)
BISB (R1)+,(SP)
    
```



CVDMAA.P11 12-DEC-80 15:59

ERROR HANDLER -- ERR6 -- VIA REGISTER ERRORS W/FULL REG. DUMP

2917	006416	012746	014021		MOV	#TXT8B,-(SP)
2918	006422	012746	012571		MOV	#FMT06A,-(SP)
2919	006426	012746	000010		MOV	#10,-(SP)
2920	006432	010600			MOV	SP,R0
2921	006434	104415			TRAP	C\$PNTX
2922	006436	062706	000022		ADD	#22,SP
2923	006442			PRINTX	#FMT06B,<B,(R1)+>,<B,(R1)+>	
2924	006442	005046			CLR	-(SP)
2925	006444	152116			BISB	(R1)+,(SP)
2926	006446	005046			CLR	-(SP)
2927	006450	152116			BISB	(R1)+,(SP)
2928	006452	012746	012637		MOV	#FMT06B,-(SP)
2929	006456	012746	000003		MOV	#3,-(SP)
2930	006462	010600			MOV	SP,R0
2931	006464	104415			TRAP	C\$PNTX
2932	006466	062706	000010		ADD	#10,SP
2933	006472	012701	003216	MOV	#BT2+8.,R1	:POINT TO 2ND HALF OF XOR VALUES
2934	006476			PRINTX	#FMT06A,#TXT8C,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>	
2935	006476	005046			CLR	-(SP)
2936	006500	152116			BISB	(R1)+,(SP)
2937	006502	005046			CLR	-(SP)
2938	006504	152116			BISB	(R1)+,(SP)
2939	006506	005046			CLR	-(SP)
2940	006510	152116			BISB	(R1)+,(SP)
2941	006512	005046			CLR	-(SP)
2942	006514	152116			BISB	(R1)+,(SP)
2943	006516	005046			CLR	-(SP)
2944	006520	152116			BISB	(R1)+,(SP)
2945	006522	005046			CLR	-(SP)
2946	006524	152116			BISB	(R1)+,(SP)
2947	006526	012746	014036		MOV	#TXT8C,-(SP)
2948	006532	012746	012571		MOV	#FMT06A,-(SP)
2949	006536	012746	000010		MOV	#10,-(SP)
2950	006542	010600			MOV	SP,R0
2951	006544	104415			TRAP	C\$PNTX
2952	006546	062706	000022		ADD	#22,SP
2953	006552			PRINTX	#FMT06B,<B,(R1)+>,<B,(R1)+>	
2954	006552	005046			CLR	-(SP)
2955	006554	152116			BISB	(R1)+,(SP)
2956	006556	005046			CLR	-(SP)
2957	006560	152116			BISB	(R1)+,(SP)
2958	006562	012746	012637		MOV	#FMT06B,-(SP)
2959	006566	012746	000003		MOV	#3,-(SP)
2960	006572	010600			MOV	SP,R0
2961	006574	104415			TRAP	C\$PNTX
2962	006576	062706	000010		ADD	#10,SP
2963	006602	012601		MOV	(SP)+,R1	:RESTORE R1
2964	006604	004737	012072	JSR	PC,NULERR	:USE COMMON ROUTINE TO TERMINATE ERROR MESSAGE
2965	006610			ENDMSG		
2966	006610					
2967	006610	104423				
2968						
2969						
2970						
2971	006612					
2972	006612					

-----  
:SBTTL ERROR HANDLER -- ERR7 -- VIA REGISTER ERRORS  
:-----  
BGNMSG ERR7  
L10011: TRAP C\$MSG  
ERR7::

CVDMAA.P11 12-DEC-80 15:59

## ERROR HANDLER -- ERR7 -- VIA REGISTER ERRORS

```

2973 006612 113701 002334      MOVB   REGNUM,R1
2974 006616 006301              ASL    R1              ;AS PASSED, THIS WAS A BYTE OFFSET
2975 006620                    PRINTB  #FMT07,#TXTVR,TXTVRT(R1)
2976 006620 016146 017554              MOV    TXTVRT(R1),-(SP)
2977 006624 012746 014327              MOV    #TXTVR,-(SP)
2978 006630 012746 012530              MOV    #FMT07,-(SP)
2979 006634 012746 000003              MOV    #3,-(SP)
2980 006640 010600                    MOV    SP,R0
2981 006642 104414                    TRAP   C$PNTB
2982 006644 062706 000010              ADD    #10,SP
2983 006650 004737 011276              JSR    PC,XORGB
2984 006654                    PRINTB  #FMT02A,<B,GDATA>,<B,BDATA>,<B,XDATA>
2985 006654 005046                    CLR    -(SP)
2986 006656 153716 002314              BISB   XDATA,(SP)
2987 006662 005046                    CLR    -(SP)
2988 006664 153716 002312              BISB   BDATA,(SP)
2989 006670 005046                    CLR    -(SP)
2990 006672 153716 002310              BISB   GDATA,(SP)
2991 006676 012746 012161              MOV    #FMT02A,-(SP)
2992 006702 012746 000004              MOV    #4,-(SP)
2993 006706 010600                    MOV    SP,R0
2994 006710 104414                    TRAP   C$PNTB
2995 006712 062706 000012              ADD    #12,SP
2996 006716 004737 012072              JSR    PC,NULERR      ;USE COMMON ROUTINE TO TERMINATE ERROR MESSAGE
2997 006722                    ENDMSG
2998 006722
2999 006722 104423                    L10012: TRAP   C$MSG
3000
3001 :-----:
3002 : SBTTL  ERROR HANDLER -- ERR47 -- FOR RAM DATA ERRORS IN STATIC TEST(S)
3003 :-----:
3003 006724                    BGNMSG  ERR47
3004 006724
3005 :
3006 : PRINT HEADING LINE # 1
3007 006724 013700 002444              MOV    TMP2,R0          ;GET TEST PATTERN CODE
3008 006730 001404                    BEQ    2$              ;ZERO IS UNDEFINED BUT THERE IS TEXT TO SAY THAT
3009 006732 020027 000006              CMP    R0,#6          ;THIS IT ALL WE UNDERSTAND FOR NOW
3010 006736 003401                    BLE    2$              ;IF WITHIN LIMITS, LET IT GO
3011 006740 005000                    CLR    R0              ;ELSE, MAKE IT 0 FOR 'UNDEFINED'
3012 006742 006300                    2$: ASL    R0              ;CONVERT TO A WORD INDEX
3013 006744 016000 007614              MOV    TXT47P(R0),R0   ;GET ADDRESS OF REQUIRED TEXT
3014 006750                    PRINTX  #FMT47A,R0     ;IDENTIFY TEST PATTERN BEING USED
3015 006750 010046                    MOV    R0,-(SP)
3016 006752 012746 007232              MOV    #FMT47A,-(SP)
3017 006756 012746 000002              MOV    #2,-(SP)
3018 006762 010600                    MOV    SP,R0
3019 006764 104415                    TRAP   C$PNTX
3020 006766 062706 000006              ADD    #6,SP
3021 :
3022 : PRINT HEADING LINE # 2
3023 006772                    PRINTX  #FMT47B          ;STANDARD PORTION OF LINE 2
3024 006772 012746 007263              MOV    #FMT47B,-(SP)
3025 006776 012746 000001              MOV    #1,-(SP)
3026 007002 010600                    MOV    SP,R0
3027 007004 104415                    TRAP   C$PNTX
3028 007006 062706 000004              ADD    #4,SP

```



CVDMAA.P11

12-DEC-80 15:59

ERROR HANDLER -- ERR47 -- FOR RAM DATA ERRORS IN STATIC TEST(S)

```

3029 ; PRINT HEADING LINE # 3
3030
3031 007012 PRINTX #FMT47C ;STANDARD PORTION OF LINE 3
3032 007012 012746 007322 MOV #FMT47C,-(SP)
3033 007016 012746 000001 MOV #1,-(SP)
3034 007022 010600 MOV SP,R0
3035 007024 104415 TRAP C$PNTX
3036 007026 062706 000004 ADD #4,SP
3037 ; PRINT HEADING LINE # 4
3038
3039 007032 PRINTX #FMT47E ;STANDARD PORTION OF LINE 4
3040 007032 012746 007350 MOV #FMT47E,-(SP)
3041 007036 012746 000001 MOV #1,-(SP)
3042 007042 010600 MOV SP,R0
3043 007044 104415 TRAP C$PNTX
3044 007046 062706 000004 ADD #4,SP
3045 ; GO PRINT DATA PORTION OF ERROR MESSAGE
3046
3047 007052 PRINTX #NEWLIN ;TERMINATE HEADER & CAUSE 1 BLANK LINE
3048 007052 012746 012121 MOV #NEWLIN,-(SP)
3049 007056 012746 000001 MOV #1,-(SP)
3050 007062 010600 MOV SP,R0
3051 007064 104415 TRAP C$PNTX
3052 007066 062706 000004 ADD #4,SP
3053 007072 005037 007104 CLR ER47CT ;RE-INITIALIZE THE DATA LINE COUNTER
3054 007076 004737 007110 JSR PC,ERR47. ;USE COMMON SUBROUTINE TO REPORT DATA
3055 007102
3056 007102
3057 007102 104423 L10013: TRAP C$MSG
3058
3059 007104 000000 ER47CT: .WORD 0 ;THIS VARIABLE WILL COUNT THE DATA LINES
3060 007106 000020 ER47MX: .WORD 16. ;THIS CONSTANT LIMITS THE DATA LINES PRINTED
3061
3062 007110 ERR47.:
3063
3064 007110 023737 007104 007106 CMP ER47CT,ER47MX ;HAVE WE REPORTED ENOUGH OF THESE DATA LINES?
3065 007116 103044 BHIS 60$ ;YES, BYPASS THIS WHOLE ROUTINE AND EXIT
3066 007120 005237 007104 INC ER47CT ;NO, COUNT THIS LINE
3067
3068 007124 113701 002450 MOVB TMP4,R1 ;GET EXPECTED DATA
3069 007130 113703 002452 MOVB TMP5,R3 ;SETUP TO CALCULATE XOR
3070 007134 074103 XOR R1,R3 ;CALCULATE XOR OF EXPECTED & ACTUAL DATA
3071 007136 PRINTX #FMT47G,TMP4,<B,R1>,<B,TMP5>,<B,R3> ;PRINT DATA LINE
3072 007136 005046 CLR -(SP)
3073 007140 150316 BISB R3,(SP)
3074 007142 005046 CLR -(SP)
3075 007144 153716 002452 BISB TMP5,(SP)
3076 007150 005046 CLR -(SP)
3077 007152 150116 BISB R1,(SP)
3078 007154 013746 002464 MOV TMP4,-(SP)
3079 007160 012746 007407 MOV #FMT47G,-(SP)
3080 007164 012746 000005 MOV #5,-(SP)
3081 007170 010600 MOV SP,R0
3082 007172 104415 TRAP C$PNTX
3083 007174 062706 000014 ADD #14,SP
3084 007200 023737 007104 007106 CMP ER47CT,ER47MX ;IF THESE TWO ARE EQUAL, WE WON'T BE PRINTING

```

CVDMAA.P11 12-DEC-80 15:59

ERROR HANDLER -- ERR47 -- FOR RAM DATA ERRORS IN STATIC TEST(S)

```

3085 007206 001010          BNE      60$          ;ANY MORE LINES FOR A WHILE. SO,
3086 007210                PRINTX  #FMT48I      ; PUT OUT A MESSAGE TO THAT EFFECT.
3087 007210 012746 010644          MOV      #FMT48I,-(SP)
3088 007214 012746 000001          MOV      #1,-(SP)
3089 007220 010600                MOV      SP,R0
3090 007222 104415                TRAP    C$PNTX
3091 007224 062706 000004          ADD     #4,SP
3092 007230 000207
3093
3094
007232 047045 051445 022462      .NLIST  BEX
007263      045 022516 031123      FMT47A: .ASCIZ  \XN%S2%ATEST PATTERN: XT\
007322 047045 051445 022463      FMT47B: .ASCIZ  \XN%S2%A (ALL VALUES IN OCTAL)\
007350 047045 051445 022463      FMT47C: .ASCIZ  \XN%S3%A RAM SHOULD\
007407      045 022516 032123      FMT47E: .ASCIZ  \XN%S3%ADDRESS BE IS XOR\
007442 046101 020114 047117      FMT47G: .ASCIZ  \XN%S4%04%S4%03%S3%03%S2%03\
007453      101 046114 055040      TXT47C: .ASCIZ  \ALL ONES\
007466 020061 044502 020124      TXT47D: .ASCIZ  \ALL ZEROES\
007510 020062 044502 051524      TXT47E: .ASCIZ  \1 BIT ALTERNATING\
007533      101 042104 042522      TXT47F: .ASCIZ  \2 BITS ALTERNATING\
007556 047111 051103 046505      TXT47G: .ASCIZ  \ADDRESS IN ADDRESS\
                                TXT47H: .ASCIZ  \INCREMENTAL VALUE IN ADDRESS\
                                .LIST  BEX
                                .EVEN
3095 007614 007614                TXT47P: .WORD   TXTML6,TXT47C,TXT47D,TXT47E,TXT47F,TXT47G,TXT47H
3096 007614 014235 007442 007453
3097 007622 007466 007510 007533
3098 007630 007556
3099
3100 ;      'TXTML6' ABOVE IS DEFINED AS 'UNDEFINED' IN THE M-LOOP FUNCTION DEF'S.
3101
3102
3103 ;-----
3104 ;.SBTTL ERROR HANDLER -- ERR48 -- FOR DATA ERRORS IN 'MOVING INVERSIONS TEST'
3105 ;-----
3105 007632          BGNMSG  ERR48
3106 007632          ERR48::
3107 ;      PRINT HEADING LINE # 1
3108
3109 007632          PRINTX  #FMT48A          ;STANDARD PORTION OF LINE 1
3110 007632 012746 010300          MOV      #FMT48A,-(SP)
3111 007636 012746 000001          MOV      #1,-(SP)
3112 007642 010600                MOV      SP,R0
3113 007644 104415                TRAP    C$PNTX
3114 007646 062706 000004          ADD     #4,SP
3115 007652 032737 000004 002350      BIT      #BIT2,PFLAG          ;IF EXTENDED INFORMATION REQUESTED,
3116 007660 001410                BEQ     2$
3117 007662          PRINTX  #FMT48B          ;PRINT EXTENDED PORTION OF LINE 1
3118 007662 012746 010347          MOV      #FMT48B,-(SP)
3119 007666 012746 000001          MOV      #1,-(SP)
3120 007672 010600                MOV      SP,R0
3121 007674 104415                TRAP    C$PNTX
3122 007676 062706 000004          ADD     #4,SP
3123 ;      PRINT HEADING LINE # 2
3124
3125 2$:          PRINTX  #FMT48C          ;STANDARD PORTION OF LINE 2
3126 007702 012746 010402          MOV      #FMT48C,-(SP)
3127 007706 012746 000001          MOV      #1,-(SP)
3128 007712 010600                MOV      SP,R0

```



CVDMAA.P11 12-DEC-80 15:59

ERROR HANDLER -- ERR48 -- FOR DATA ERRORS IN 'MOVING INVERSIONS TEST'

```

3129 007714 104415
3130 007716 062706 000004 TRAP C$PNTX
3131 ADD #4,SP
3132 : PRINT HEADING LINE # 3
3133
3134 007722 PRINTX #FMT48E ;STANDARD PORTION OF LINE 3
3135 007722 012746 010437 MOV #FMT48E,-(SP)
3136 007726 012746 000001 MOV #1,-(SP)
3137 007732 010600 MOV SP,R0
3138 007734 104415 TRAP C$PNTX
3139 007736 062706 000004 ADD #4,SP
3140 007742 032737 000004 002350 BIT #BIT2,PFLAG ;IF EXTENDED INFORMATION REQUESTED,
3141 007750 001410 BEQ 6$
3142 007752 PRINTX #FMT48F ;PRINT EXTENDED PORTION OF LINE 3
3143 007752 012746 010506 MOV #FMT48F,-(SP)
3144 007756 012746 000001 MOV #1,-(SP)
3145 007762 010600 MOV SP,R0
3146 007764 104415 TRAP C$PNTX
3147 007766 062706 000004 ADD #4,SP
3148 : GO PRINT DATA PORTION OF ERROR MESSAGE
3149
3150 007772 6$: PRINTX #NEWLIN ;TERMINATE HEADER & CAUSE 1 BLANK LINE
3151 007772 012746 012121 MOV #NEWLIN,-(SP)
3152 007776 012746 000001 MOV #1,-(SP)
3153 010002 010600 MOV SP,R0
3154 010004 104415 TRAP C$PNTX
3155 010006 062706 000004 ADD #4,SP
3156 010012 005037 010024 CLR ER48CT ;RE-INITIALIZE THE DATA LINE COUNTER
3157 010016 004737 010030 JSR PC,ERR48. ;USE COMMON SUBROUTINE TO REPORT DATA
3158 010022 ENDMSG
3159 010022
3160 010022 104423 L10014: TRAP C$MSG
3161
3162 010024 000000 ER48CT: .WORD 0 ;THIS VARIABLE WILL COUNT THE DATA LINES
3163 010026 000020 ER48MX: .WORD 16. ;THIS CONSTANT LIMITS THE DATA LINES PRINTED
3164
3165 010030 ERR48.:
3166
3167 010030 023737 010024 010026 CMP ER48CT,ER48MX ;HAVE WE REPORTED ENOUGH OF THESE DATA LINES?
3168 010036 103117 BHIS 60$ ;YES, BYPASS THIS WHOLE ROUTINE AND EXIT
3169 010040 005237 010024 INC ER48CT ;NO, COUNT THIS LINE
3170
3171 : DETERMINT WHICH ERROR CALL GOT US HERE -- PRE-WRITE OR POST-WRITE:
3172
3173 010044 032737 000002 002476 BIT #BIT1,TMPF ;DID PRE-WRITE ERROR CALL GET US HERE?
3174 010052 001405 BEQ 2$ ;NO, THEN SETUP FOR 'POST' IN ERROR MESSAGE
3175 010054 012700 010736 MOV #TXT48A,R0 ;YES, SETUP FOR 'PRE' IN ERROR MESSAGE
3176 010060 113701 002450 MOVB TMP4,R1 ;GET EXPECTED DATA (BEFORE WRITING NEW VALUE)
3177 010064 000404 BR 4$
3178
3179 010066 012700 010743 2$: MOV #TXT48B,R0 ;POINT TO 'POST' TEXT
3180 010072 113701 002451 MOVB TMP4+1,R1 ;GET EXPECTED DATA (AFTER WRITING NEW VALUE)
3181 010076 013703 002452 4$: MOV TMP5,R3 ;SETUP TO CALCULATE XOR
3182 010102 074103 XOR R1,R3 ;CALCULATE XOR OF EXPECTED & ACTUAL DATA
3183 010104 PRINTX #FMT48G,R0,TMPA,<B,R1>,<B,TMP5>,<B,R3> ;PRINT STANDARD DATA LINE
3184 010104 005046 CLR -(SP)

```

CVDMAA.P11 12-DEC-80 15:59

ERROR HANDLER -- ERR48 -- FOR DATA ERRORS IN 'MOVING INVERSIONS TEST'

```

3185 010106 150316
3186 010110 005046
3187 010112 153716 002452
3188 010116 005046
3189 010120 150116
3190 010122 013746 002464
3191 010126 010046
3192 010130 012746 010551
3193 010134 012746 000006
3194 010140 010600
3195 010142 104415
3196 010144 062706 000016
3197 010150 032737 000004 002350
3198 010156 001433
3199
3200 010160 013701 002470
3201 010164 042701 177776
3202 010170 005737 002472
3203 010174 001003
3204 010176 012700 010750
3205 010202 000402
3206 010204 012700 010755 6$:
3207 010210 8$:
3208 010210 005046
3209 010212 153716 002462
3210 010216 010046
3211 010220 010146
3212 010222 013746 002466
3213 010226 012746 010611
3214 010232 012746 000005
3215 010236 010600
3216 010240 104415
3217 010242 062706 000014
3218 010246 023737 010024 010026 10$:
3219 010254 001010
3220 010256
3221 010256 012746 010644
3222 010262 012746 000001
3223 010266 010600
3224 010270 104415
3225 010272 062706 000004
3226 010276 60$:
3227 010276 000207
3228

010300 047045 051445 022462 .NLIST
010347 045 032523 040445 BEX
010402 047045 051445 022463 FMT48A: .ASCIZ \N%2%APRE OR (ALL VALUES IN OCTAL)\
010437 045 022516 031123 FMT48B: .ASCIZ \S5%EXTENDED INFORMATION:\
010506 051445 022465 041101 FMT48C: .ASCIZ \N%3%APOST RAM SHOULD\
010551 045 022516 031523 FMT48E: .ASCIZ \N%2%WRITE ADDRESS BE IS XOR\
010611 045 033123 047445 FMT48F: .ASCIZ \S5%BIT DATA SEQ LSB(DECIMAL)\
010644 047045 047045 051445 FMT48G: .ASCIZ \N%3%T%4%0%4%4%0%3%3%0%3%2%0%3\
010736 051120 020105 000 TXT48H: .ASCIZ \S6%01%5%01%5%3%T%2%D2%A.\
010743 120 051517 000124 FMT48I: .ASCIZ \N%N%5%AFURTHER DATA LINES SUPRESSED UNTIL NEW TEST DATA\
010750 043040 042127 000 TXT48A: .ASCIZ \PRE \
010755 102 053513 000104 TXT48B: .ASCIZ \POST\
TXT48C: .ASCIZ \ FWD\
TXT48D: .ASCIZ \BKWD\

```

```

BISB R3,(SP)
CLR -(SP)
BISB TMP5,(SP)
CLR -(SP)
BISB R1,(SP)
MOV TMPA,-(SP)
MOV RO,-(SP)
MOV #FMT48G,-(SP)
MOV #6,-(SP)
MOV SP,RO
TRAP C$PNTX
ADD #16,SP
;IF EXTENDED INFORMATION REQUESTED,
;SETUP FOR PRINTING OF EXTENDED INFORMATION
;DATA BIT VALUE (0 OR 1)
; MAKE SURE WE ONLY HAVE ONE BIT
;DIRECTION?
;BACKWARD --
;FORWARD ---
;BACKWARD --
;PRINT EXTENDED INFORMATION
CLR -(SP)
BISB TMP9,(SP)
MOV RO,-(SP)
MOV R1,-(SP)
MOV TMPB,-(SP)
MOV #FMT48H,-(SP)
MOV #5,-(SP)
MOV SP,RO
TRAP C$PNTX
ADD #14,SP
;IF THESE TWO ARE EQUAL, WE WON'T BE PRINTING
;ANY MORE LINES FOR A WHILE. SO,
; PUT OUT A MESSAGE TO THAT EFFECT.
MOV #FMT48I,-(SP)
MOV #1,-(SP)
MOV SP,RO
TRAP C$PNTX
ADD #4,SP

```



CVDMAA.P11 12-DEC-80 15:59

ERROR HANDLER -- ERR48 -- FOR DATA ERRORS IN 'MOVING INVERSIONS TEST'

.LIST BEX  
.EVEN

-----  
:SBTTL ERROR HANDLER -- ERR50 -- FOR REPORTING TIMER # 1 ERRORS  
-----

```

3229
3230
3231
3232
3233
3234
3235 010762
3236 010762
3237 010762 010146
3238 010764 113701 002467
3239 010770 000241
3240
3241 010772 042701 177477
3242 010776 106101
3243 011000 106101
3244 011002 106101
3245
3246
3247
3248 011004
3249 011004 010146
3250 011006 012746 012747
3251 011012 012746 000002
3252 011016 010600
3253 011020 104415
3254 011022 062706 000006
3255
3256
3257 011026
3258 011026 012746 013021
3259 011032 012746 000001
3260 011036 010600
3261 011040 104415
3262 011042 062706 000004
3263
3264
3265 011046
3266 011046 005046
3267 011050 153716 002455
3268 011054 005046
3269 011056 153716 002457
3270 011062 005046
3271 011064 153716 002451
3272 011070 005046
3273 011072 153716 002453
3274 011076 012746 014053
3275 011102 012746 013102
3276 011106 012746 000006
3277 011112 010600
3278 011114 104415
3279 011116 062706 000016
3280 011122
3281 011122 005046
3282 011124 153716 002475
3283 011130 005046

```

BGNMSG ERR50

ERR50::

```

MOV R1,-(SP) ;SAVE R1 FOR CALLER
MOVB TMPB+1,R1 ;GET THE MODE LAST SETUP
CLC ;SEEING AS THE CARRY BIT WILL BE ROTATED INTO
;THE DATA, WE HAD BETTER CLEAR IT JUST IN CASE.
BIC #^C<BIT6+BIT7>,R1 ;LOOK @ JUST THE TIMER 1 MODE DEFINITION
ROLB R1 ;POSITION IT FOR PRINTOUT
ROLB R1
ROLB R1

```

;IDENTIFY THE MODE BEING USED AT THE TIME:

PRINTX #FMT50A,R1

```

MOV R1,-(SP)
MOV #FMT50A,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C$PNTX
ADD #6,SP

```

;PRINT THE HEADING TO IDENTIFY THE REGISTERS:

PRINTX #FMT50B

```

MOV #FMT50B,-(SP)
MOV #1,-(SP)
MOV SP,R0
TRAP C$PNTX
ADD #4,SP

```

;AND THE VALUES THAT WERE LOADED INTO THE REGISTERS:

PRINTX #FMT50C,<B,TMP5+1>,<B,TMP4+1>,<B,TMP7+1>,<B,TMP6+1>

```

CLR -(SP)
BISB TMP6+1,(SP)
CLR -(SP)
BISB TMP7+1,(SP)
CLR -(SP)
BISB TMP4+1,(SP)
CLR -(SP)
BISB TMP5+1,(SP)
MOV #TXT8D,-(SP)
MOV #FMT50C,-(SP)
MOV #6,-(SP)
MOV SP,R0
TRAP C$PNTX
ADD #16,SP

```

PRINTX #FMT50D,<B,TMPB+1>,<B,TMPE+1>

```

CLR -(SP)
BISB TMPE+1,(SP)
CLR -(SP)

```

CVDMAA.P11 12-DEC-80 15:59

ERROR HANDLER -- ERR50 -- FOR REPORTING TIMER # 1 ERRORS

```

3284 011132 153716 002467
3285 011136 012746 013142
3286 011142 012746 000003
3287 011146 010600
3288 011150 104415
3289 011152 062706 000010
3290
3291
3292 011156
3293 011156 005046
3294 011160 153716 002454
3295 011164 005046
3296 011166 153716 002456
3297 011172 005046
3298 011174 153716 002450
3299 011200 005046
3300 011202 153716 002452
3301 011206 012746 014070
3302 011212 012746 013102
3303 011216 012746 000006
3304 011222 010600
3305 011224 104415
3306 011226 062706 000016
3307 011232
3308 011232 005046
3309 011234 153716 002472
3310 011240 005046
3311 011242 153716 002466
3312 011246 012746 013157
3313 011252 012746 000003
3314 011256 010600
3315 011260 104415
3316 011262 062706 000010
3317
3318 011266 004737 012072
3319 011272 012601
3320 011274
3321 011274
3322 011274 104423
3323
3324
    
```

;AND THE VALUES READ FROM THOSE REGISTERS:

PRINTX #FMT50C,#TXT8E,<B,TMP5>,<B,TMP4>,<B,TMP7>,<B,TMP6>

PRINTX #FMT50E,<B,TMPB>,<B,TMPD>

JSR PC,MULERR  
MOV (SP)+,R1  
ENDMSG

;USE COMMON ROUTINE TO TERMINATE ERROR MESSAGE  
;RESTORE R1 FOR CALLER

L10015:

```

BISB TMPB+1,(SP)
MOV #FMT50D,-(SP)
MOV #3,-(SP)
MOV SP,R0
TRAP C$PNTX
ADD #10,SP

CLR -(SP)
BISB TMP6,(SP)
CLR -(SP)
BISB TMP7,(SP)
CLR -(SP)
BISB TMP4,(SP)
CLR -(SP)
BISB TMP5,(SP)
MOV #TXT8E,-(SP)
MOV #FMT50C,-(SP)
MOV #6,-(SP)
MOV SP,R0
TRAP C$PNTX
ADD #16,SP

CLR -(SP)
BISB TMPD,(SP)
CLR -(SP)
BISB TMPB,(SP)
MOV #FMT50E,-(SP)
MOV #3,-(SP)
MOV SP,R0
TRAP C$PNTX
ADD #10,SP

TRAP C$MSG
    
```



CVDMAA.P11 12-DEC-80 15:59

## ERROR HANDLER SUBROUTINES

```

3325 .SBTTL ERROR HANDLER SUBROUTINES
3326 -----
3327 ----- SUBROUTINES USED ONLY BY ERROR HANDLERS -----
3328 -----
3329 -----
3330 -----
3331 .SBTTL ERROR HANDLER SUBROUTINE -- XORGB
3332 -----
3333 PERFORM EXCLUSIVE OR BETWEEN 'GDATA' & 'BDATA' PUTTING
3334 THE RESULT IN 'XDATA'
3335 -----
3336 XORGB: MOV R1,-(SP) ;PRESERVE WORKING REGISTER
3337 MOV GDATA,R1 ;GET 'GOOD' DATA
3338 MOV BDATA,XDATA ;AND 'BAD' DATA
3339 XOR R1,XDATA ;PERFORM EXCLUSIVE OR
3340 MOV (SP)+,R1 ;RESTORE R1
3341 RTS PC ;RETURN
3342 -----
3343 -----
3344 -----
3345 .SBTTL ERROR HANDLER SUBROUTINE -- ERR4$
3346 -----
3347 IDENTIFY & DUMP THE BYTE SELECT REGISTERS
3348 -----
3349 ERR4$: PRINTX #FMT4,#TXT3,#TXT1
3350 MOV #TXT1,-(SP)
3351 MOV #TXT3,-(SP)
3352 MOV #FMT4,-(SP)
3353 MOV #3,-(SP)
3354 MOV SP,R0
3355 TRAP C$PNTX
3356 ADD #10,SP
3357 PRINTX #FMT4A,<B,BSR0>,<B,BSR1>,<B,BSR2>,<B,BSR3>
3358 CLR -(SP)
3359 BISB BSR3,(SP)
3360 CLR -(SP)
3361 BISB BSR2,(SP)
3362 CLR -(SP)
3363 BISB BSR1,(SP)
3364 CLR -(SP)
3365 BISB BSR0,(SP)
3366 MOV #FMT4A,-(SP)
3367 MOV #5,-(SP)
3368 MOV SP,R0
3369 TRAP C$PNTX
3370 ADD #14,SP
3371 PRINTX #FMT4B,#TXT2
3372 MOV #TXT2,-(SP)
3373 MOV #FMT4B,-(SP)
3374 MOV #2,-(SP)
3375 MOV SP,R0
3376 TRAP C$PNTX
3377 ADD #6,SP
3378 PRINTX #FMT4C,<B,BSR4>,<B,BSR5>,<B,BSR6>,<B,BSR7>
3379 CLR -(SP)
3380 BISB BSR7,(SP)

```

011276	010146			
011300	013701	002310		
011304	013737	002312	002314	
011312	074137	002314		
011316	012601			
011320	000207			
011322				
011322	012746	013253		
011326	012746	013454		
011332	012746	012245		
011336	012746	000003		
011342	010600			
011344	104415			
011346	062706	000010		
011352				
011352	005046			
011354	153716	002254		
011360	005046			
011362	153716	002252		
011366	005046			
011370	153716	002250		
011374	005046			
011376	153716	002246		
011402	012746	012305		
011406	012746	000005		
011412	010600			
011414	104415			
011416	062706	000014		
011422				
011422	012746	013311		
011426	012746	012340		
011432	012746	000002		
011436	010600			
011440	104415			
011442	062706	000006		
011446				
011446	005046			
011450	153716	002264		

CVDMAA.P11 12-DEC-80 15:59

ERROR HANDLER SUBROUTINE -- ERR4\$

3381	011454	005046			CLR	-(SP)
3382	011456	153716	002262		BISB	BSR6,(SP)
3383	011462	005046			CLR	-(SP)
3384	011464	153716	002260		BISB	BSR5,(SP)
3385	011470	005046			CLR	-(SP)
3386	011472	153716	002256		BISB	BSR4,(SP)
3387	011476	012746	012345		MOV	#FMT4C,-(SP)
3388	011502	012746	000005		MOV	#5,-(SP)
3389	011506	010600			MOV	SP,R0
3390	011510	104415			TRAP	CSPNTX
3391	011512	062706	000014		ADD	#14,SP
3392	011516			PRINTX	#FMT4B,#TXT2A	
3393	011516	012746	013353		MOV	#TXT2A,-(SP)
3394	011522	012746	012340		MOV	#FMT4B,-(SP)
3395	011526	012746	000002		MOV	#2,-(SP)
3396	011532	010600			MOV	SP,R0
3397	011534	104415			TRAP	CSPNTX
3398	011536	062706	000006		ADD	#6,SP
3399	011542			PRINTX	#FMT4A,<B,BSR10>,<B,BSR11>,<B,BSR12>,<B,BSR13>	
3400	011542	005046			CLR	-(SP)
3401	011544	153716	002274		BISB	BSR13,(SP)
3402	011550	005046			CLR	-(SP)
3403	011552	153716	002272		BISB	BSR12,(SP)
3404	011556	005046			CLR	-(SP)
3405	011560	153716	002270		BISB	BSR11,(SP)
3406	011564	005046			CLR	-(SP)
3407	011566	153716	002266		BISB	BSR10,(SP)
3408	011572	012746	012305		MOV	#FMT4A,-(SP)
3409	011576	012746	000005		MOV	#5,-(SP)
3410	011602	010600			MOV	SP,R0
3411	011604	104415			TRAP	CSPNTX
3412	011606	062706	000014		ADD	#14,SP
3413	011612			PRINTX	#FMT4B,#TXT2B	
3414	011612	012746	013412		MOV	#TXT2B,-(SP)
3415	011616	012746	012340		MOV	#FMT4B,-(SP)
3416	011622	012746	000002		MOV	#2,-(SP)
3417	011626	010600			MOV	SP,R0
3418	011630	104415			TRAP	CSPNTX
3419	011636	062706	000006		ADD	#6,SP
3420	011636			PRINTX	#FMT4C,<B,BSR14>,<B,BSR15>,<B,BSR16>,<B,BSR17>	
3421	011636	005046			CLR	-(SP)
3422	011640	153716	002304		BISB	BSR17,(SP)
3423	011644	005046			CLR	-(SP)
3424	011646	153716	002302		BISB	BSR16,(SP)
3425	011652	005046			CLR	-(SP)
3426	011654	153716	002300		BISB	BSR15,(SP)
3427	011660	005046			CLR	-(SP)
3428	011662	153716	002276		BISB	BSR14,(SP)
3429	011666	012746	012345		MOV	#FMT4C,-(SP)
3430	011672	012746	000005		MOV	#5,-(SP)
3431	011676	010600			MOV	SP,R0
3432	011700	104415			TRAP	CSPNTX
3433	011702	062706	000014		ADD	#14,SP
3434	011706	000207		RTS	PC	
3435						
3436						



CVDMAA.P11 12-DEC-80 15:59

.....ERROR HANDLER SUBROUTINE -- ERR5\$

3437  
 3438  
 3439  
 3440 011710  
 3441 011710  
 3442 011710 012746 013504  
 3443 011714 012746 013607  
 3444 011720 012746 012245  
 3445 011724 012746 000003  
 3446 011730 010600  
 3447 011732 104415  
 3448 011734 062706 000010  
 3449 011740  
 3450 011740 013746 002254  
 3451 011744 013746 002252  
 3452 011750 013746 002250  
 3453 011754 013746 002246  
 3454 011760 012746 012730  
 3455 011764 012746 000005  
 3456 011770 010600  
 3457 011772 104415  
 3458 011774 062706 000014  
 3459 012000  
 3460 012000 012746 013544  
 3461 012004 012746 012340  
 3462 012010 012746 000002  
 3463 012014 010600  
 3464 012016 104415  
 3465 012020 062706 000006  
 3466 012024  
 3467 012024 013746 002264  
 3468 012030 013746 002262  
 3469 012034 013746 002260  
 3470 012040 013746 002256  
 3471 012044 012746 012730  
 3472 012050 012746 000005  
 3473 012054 010600  
 3474 012056 104415  
 3475 012060 062706 000014  
 3476 012064 004737 012072  
 3477 012070 000207  
 3478  
 3479  
 3480  
 3481  
 3482 012072  
 3483 012072 012746 012114  
 3484 012076 012746 000001  
 3485 012102 010600  
 3486 012104 104414  
 3487 012106 062706 000004  
 3488 012112 000207  
 3489

.SBTTL .....ERROR HANDLER SUBROUTINE -- ERR5\$

.....COMMON ERROR SUBROUTINE TO PRINT SELECT REGISTERS

ERR5\$:

PRINTX #FMT4,#TXT6,#TXT4

MOV #TXT4,-(SP)  
 MOV #TXT6,-(SP)  
 MOV #FMT4,-(SP)  
 MOV #3,-(SP)  
 MOV SP,R0  
 TRAP CSPNTX  
 ADD #10,SP

PRINTX #FMT11,WSR0,WSR2,WSR4,WSR6 ;DUMP THE SELECT REGISTERS

MOV WSR6,-(SP)  
 MOV WSR4,-(SP)  
 MOV WSR2,-(SP)  
 MOV WSR0,-(SP)  
 MOV #FMT11,-(SP)  
 MOV #5,-(SP)  
 MOV SP,R0  
 TRAP CSPNTX  
 ADD #14,SP

PRINTX #FMT4B,#TXT4A

MOV #TXT4A,-(SP)  
 MOV #FMT4B,-(SP)  
 MOV #2,-(SP)  
 MOV SP,R0  
 TRAP CSPNTX  
 ADD #6,SP

PRINTX #FMT11,WSR10,WSR12,WSR14,WSR16

MOV WSR16,-(SP)  
 MOV WSR14,-(SP)  
 MOV WSR12,-(SP)  
 MOV WSR10,-(SP)  
 MOV #FMT11,-(SP)  
 MOV #5,-(SP)  
 MOV SP,R0  
 TRAP CSPNTX  
 ADD #14,SP

JSR PC,NULERR ;USE COMMON ROUTINE TO TERMINATE ERROR MESSAGE  
 RTS PC

.SBTTL SUBROUTINE TO PERFORM 'PRINTB #ENDEMB'

NULERR: PRINTB #ENDEMB ;TERMINATE ERROR MESSAGE

MOV #ENDEMB,-(SP)  
 MOV #1,-(SP)  
 MOV SP,R0  
 TRAP CSPNTB  
 ADD #4,SP

RTS PC

CVDMAA.P11 12-DEC-80 15:59

FORMAT SPEC'S FOR ERROR HANDLERS -- 'FMT\_\_\_'

3490  
3491  
3492  
3493  
3494

.SBTTL FORMAT SPEC'S FOR ERROR HANDLERS -- 'FMT\_\_\_'

----- FORMAT SPEC'S USED BY ERROR HANDLERS -----

```

.NLIST BEX
012114 047045 047045 000 ENDEMB: .ASCIZ /%N%N/
012121 045 000116 NEWLIN: .ASCIZ /%N/

012124 047045 040445 040506 FMT02: .ASCIZ /%N%AFAILING REG = %T%ASEL%02/
012161 045 022516 020101 FMT02A: .ASCIZ /%N%A EXPECTED: %03%A ACTUAL: %03%A XOR: %05%/
012245 045 022516 020101 FMT4: .ASCIZ /%N%A THE CONTENTS OF ALL %T%N%T/
012305 045 022516 030523 FMT4A: .ASCIZ /%N%S1%03%S5%03%S5%03%S5%03/
012340 047045 052045 000 FMT4B: .ASCIZ /%N%T/
012345 045 022516 032523 FMT4C: .ASCIZ /%N%S5%03%S5%03%S5%03%S5%03/
012400 047045 040445 020040 FMT5: .ASCIZ /%N%A WHEN %03%A LOADED INTO BSEL1/
012443 045 022516 020101 FMT5A: .ASCIZ /%N%A ATTEMPTING "M-LOOP" FUNCTION CODE %02%A (%T%A)/
012530 040445 020040 042504 FMT07: .ASCIZ /%A DETECTED IN %T%T%A --/
012562 047045 047045 052045 FMT06: .ASCIZ /%N%N%T/
012571 045 022516 022524 FMT06A: .ASCIZ /%N%T%03%S2%03%S2%03%S2%03%S2%03%S2%03/
012637 045 031123 047445 FMT06B: .ASCIZ /%S2%03%S2%03/
012654 047045 040445 020040 FMT10: .ASCIZ /%N%A EXPECTED: %08%A ACTUAL: %08%A XOR: %08%/
012730 047045 047445 022470 FMT11: .ASCIZ /%N%08%08%08%08/
012747 045 022516 020101 FMT50A: .ASCIZ /%N%A TIMER # 1 MODE: %01%A REGISTERS:/
013021 045 022516 030523 FMT50B: .ASCIZ /%N%S15%AT1CH T1CL T1LH T1LL ACR IFR IER/
013102 047045 051445 022463 FMT50C: .ASCIZ /%N%S3%T%S1%03%S3%03%S3%03%S3%03/
013142 051445 022463 031517 FMT50D: .ASCIZ /%S3%03%S9%03/
013157 045 031523 047445 FMT50E: .ASCIZ /%S3%03%S3%03/
013174 047045 022462 030523 FMT50M: .ASCIZ /%N2%S10%(T1CH & T1CL HAVEN'T YET BEEN LOADED)/

```

.SBTTL TEXT STRINGS FOR ERROR HANDLERS -- 'TXT\_\_\_'

----- TEXT USED BY ERROR HANDLERS -----

```

013253 102 042523 030114 TXT1: .ASCIZ /BSEL0 BSEL1 BSEL2 BSEL3/
013311 040 020040 041040 TXT2: .ASCIZ / BSEL4 BSEL5 BSEL6 BSEL7/
013353 102 042523 030514 TXT2A: .ASCIZ /BSEL10 BSEL11 BSEL12 BSEL13/
013412 020040 041040 042523 TXT2B: .ASCIZ / BSEL14 BSEL15 BSEL16 BSEL17/
013454 041040 052131 020105 TXT3: .ASCIZ / BYTE SELECT REG'S ARE:/
013504 020040 051440 046105 TXT4: .ASCIZ / SEL0 SEL2 SEL4 SEL6/
013544 020040 051440 046105 TXT4A: .ASCIZ / SEL10 SEL12 SEL14 SEL16/
013605 102 000 TXT5: .ASCIZ /B/
013607 040 042523 042514 TXT6: .ASCIZ / SELECT REG'S ARE:/
013632 051040 043505 051511 TXT7: .ASCIZ / REGISTERS ORB ORA DDRB DDRA T1CL T1CH T1LL T1LH /
013717 040 020040 020040 TXT7A: .ASCIZ / T2CL T2CH SR ACR PCR IFR IER ORA /
014004 042440 050130 041505 TXT8A: .ASCIZ / EXPECTED: /
014021 040 041501 052524 TXT8B: .ASCIZ / ACTUAL: /
014036 054040 051117 020072 TXT8C: .ASCIZ / XOR: /
014053 040 047514 042101 TXT8D: .ASCIZ / LOADED: /
014070 051040 040505 035104 TXT8E: .ASCIZ / READ: /

014105 116 050117 000 TXTML0: .ASCIZ /NOP/
014111 122 040505 020104 TXTML1: .ASCIZ /READ 1 BYTE/
014125 127 044522 042524 TXTML2: .ASCIZ /WRITE 1 BYTE/
014142 050116 026522 052517 TXTML3: .ASCIZ /NPR-OUT 256 BYTES/
014164 050116 026522 047111 TXTML4: .ASCIZ /NPR-IN 256 BYTES/

```



CVDMAA.P11 12-DEC-80 15:59

## TEXT STRINGS FOR ERROR HANDLERS -- 'TXT\_--'

014205	123	052105	046440	TXTML5:	.ASCIZ	/SET MICROPROCESSOR'S PC/
014235	125	042116	043105	TXTML6:	.ASCIZ	/UNDEFINED/
014247	123	052105	046440	TXTML7:	.ASCIZ	/SET MAINT INTR & CLR INTR DISABLE IN CPU STATUS/
014327	126	040511	051040	TXTVR:	.ASCIZ	/VIA REGISTER /
014345	117	041122	000	TXTVR0:	.ASCIZ	/ORB/
014351	117	040522	000	TXTVR1:	.ASCIZ	/ORA/
014355	104	051104	000102	TXTVR2:	.ASCIZ	/DDR8/
014362	042104	040522	000	TXTVR3:	.ASCIZ	/DDRA/
014367	124	041461	000114	TXTVR4:	.ASCIZ	/T1CL/
014374	030524	044103	000	TXTVR5:	.ASCIZ	/T1CH/
014401	124	046061	000114	TXTVR6:	.ASCIZ	/T1LL/
014406	030524	044114	000	TXTVR7:	.ASCIZ	/T1LH/
014413	124	041462	000114	TXTVR8:	.ASCIZ	/T2CL/
014420	031124	044103	000	TXTVR9:	.ASCIZ	/T2CH/
014425	123	000122		TXTVRA:	.ASCIZ	/SR/
014430	041501	000122		TXTVRB:	.ASCIZ	/ACR/
014434	041520	000122		TXTVRC:	.ASCIZ	/PCR/
014440	043111	000122		TXTVRD:	.ASCIZ	/IFR/
014444	042511	000122		TXTVRE:	.ASCIZ	/IER/
014450	051117	000101		TXTVRF:	.ASCIZ	/ORA/

CVDMAA.P11 12-DEC-80 15:59

ERROR MESSAGES -- 'EM\_\_'

.SBTTL ERROR MESSAGES -- 'EM\_\_'

----- ERROR MESSAGES USED BY ERROR CALL'S -----

014454	044515	051103	026517	EM3:	.ASCIZ	/MICRO-DIAG. FAILURE/
014500	051115	054504	052040	EM4:	.ASCIZ	/MRDY TIMEOUT/
014515	115	051501	042524	EM5:	.ASCIZ	/MASTER CLR FAILURE/
014540	051503	020122	042101	EM6:	.ASCIZ	/CSR ADDRESS FAILURE/
014564	051503	020122	040504	EM7:	.ASCIZ	/CSR DATA PAT FAILURE/
014611	102	042523	030114	EM8:	.ASCIZ	/BSELO SET=ALL ONES/
014634	054105	042524	047122	EM9:	.ASCIZ	/EXTERNAL BUS RESET FAILURE/
014657	102	042101	041440	EM14:	.ASCIZ	/BAD CSR VALUE(S) AFTER MASTER CLEAR/
014733	042	051115	054504	EM15:	.ASCIZ	/'MRDY' DIDN'T GO LOW WHILE PROCESSING A COMMAND/
015013	104	053115	051447	EM16:	.ASCIZ	/DMV'S RAM LOC. (CORRESPONDING TO BSELO) NOT PROPERLY WRITTEN/
015110	032466	031060	053440	EM17:	.ASCIZ	/6502 WRITE FUNC. FAILURE AFTER 'RUN' BIT IS SET/
015170	032466	031060	051440	EM17A:	.ASCIZ	/6502 STILL RUNNING AFTER 'RUN' BIT CLEARED/
015243	126	040511	051440	EM20:	.ASCIZ	/VIA STATIC REGISTER ERROR/
015275	126	040511	051440	EM20A:	.ASCIZ	/VIA STATIC REGISTER ERROR -- TIMER NOT RUNNING/
015355	126	040511	051440	EM20B:	.ASCIZ	/VIA STATIC REGISTER ERROR -- TIMER CROSS TALK ERROR/
015442	042522	044507	052123	EM21:	.ASCIZ	/REGISTER NOT PROPERLY ZEROED/
015477	132	051105	044517	EM22:	.ASCIZ	/ZEROING DDRB EFFECTED DDRA/
015532	042532	047522	047111	EM22A:	.ASCIZ	/ZEROING DDRA EFFECTED DDRB/
015565	122	040505	027504	EM25:	.ASCIZ	'READ/WRITE DATA ERROR'
015613	125	042516	050130	EM34:	.ASCIZ	/UNEXPECTED 'A' INTERRUPT/
015644	047125	054105	042520	EM34B:	.ASCIZ	/UNEXPECTED 'B' INTERRUPT/
015675	122	046501	042040	EM47A:	.ASCIZ	/RAM DATA ERROR ON INITIAL WRITE/
015735	122	046501	042040	EM47B:	.ASCIZ	/RAM DATA ERROR ON RE-READ AFTER TEST AREA FILLED/
016016	040522	020115	040504	EM48A:	.ASCIZ	/RAM DATA ERROR -- MOVING INVERSIONS TEST/
016067	042	030524	020042	EM50A:	.ASCIZ	\ 'T1' FLAG NOT CLEARED BY LOADING T1LH\
016135	042	030524	020042	EM50B:	.ASCIZ	\ 'T1' FLAG NOT CLEARED BY LOADING T1CH\
016203	042	030524	020042	EM50C:	.ASCIZ	\ 'T1' FLAG NOT CLEARED BY READING T1CL\
016251	126	040511	051447	EM50D:	.ASCIZ	\ VIA'S T1CL NOT DECREMENTING\
016305	126	040511	051447	EM50E:	.ASCIZ	\ VIA'S T1CH NOT DECREMENTING\
016341	042	030524	020042	EM50F:	.ASCIZ	\ 'T1' FLAG NOT SET ON TIMER 1 TIMEOUT\
016406	052042	021061	043040	EM50G:	.ASCIZ	\ 'T1' FLAG CLEARED BY READING T1CH\
016450	044526	023501	020123	EM50H:	.ASCIZ	\ VIA'S T1LL IMPROPERLY LOADED BY WRITING T1CL @ ADDR 4\
016536	052042	021061	043040	EM50I:	.ASCIZ	\ 'T1' FLAG CLEARED BY READING T1LL\
016600	044526	023501	020123	EM50J:	.ASCIZ	\ VIA'S T1LH IMPROPERLY LOADED BY WRITING T1CH @ ADDR 5\
016666	052042	021061	043040	EM50K:	.ASCIZ	\ 'T1' FLAG CLEARED BY READING T1LH\
016730	052042	021061	043040	EM50L:	.ASCIZ	\ 'T1' FLAG NOT SET AFTER RE-LOADING T1CH & TIMEOUT\
017012	052042	021061	043040	EM50M:	.ASCIZ	\ 'T1' FLAG CLEARED BY LOADING T1LL\
017054	052042	021061	043040	EM50N:	.ASCIZ	\ 'T1' FLAG NOT CLEARED BY LOADING T1CH\
017122	050042	033502	020042	EM50S:	.ASCIZ	\ 'PB7' W/IN VIA NOT SET ON TIMER 1 TIMEOUT\
017174	050042	033502	020042	EM50U:	.ASCIZ	\ 'PB7' NOT SET AFTER TIMER 1 TIMEOUT\
017240	050042	033502	020042	EM50V:	.ASCIZ	\ 'PB7' NOT DRIVEN LOW BY LOADING T1CH\
017305	042	041120	021067	EM50W:	.ASCIZ	\ 'PB7' UNEXPECTEDLY MODIFIED BY TIMER 1\
017354	052042	021061	047040	EM50X:	.ASCIZ	\ 'T1' NOT RESET AFTER BEING CLEARED\
017417	042	041120	021067	EM50Y:	.ASCIZ	\ 'PB7' PREMATURELY SET DURING T1 COUNTDOWN\
017471	042	041120	021067	EM50Z:	.ASCIZ	\ 'PB7' NOT SET AFTER SECOND CYCLE\

.EVEN



CVDMAA.P11 12-DEC-80 15:59

TEXT ADDRESS TABLES FOR ERROR HANDLERS -- 'TXT\_\_T'

.SBTTL TEXT ADDRESS TABLES FOR ERROR HANDLERS -- 'TXT\_\_T'

----- TEXT ADDRESS TABLES USED BY ERROR HANDLERS -----

017532 014105 014111 014125 TXTMLT: .WORD TXTML0,TXTML1,TXTML2,TXTML3,TXTML4,TXTML5,TXTML6,TXTML7

017552 014327  
017554 014345 014351 014355 TXTVRT: .WORD TXTVR  
017574 014413 014420 014425 .WORD TXTVR0,TXTVR1,TXTVR2,TXTVR3,TXTVR4,TXTVR5,TXTVR6,TXTVR7  
.WORD TXTVR8,TXTVR9,TXTVRA,TXTVRB,TXTVRC,TXTVRD,TXTVRE,TXTVRF

.LIST BEX

CVDMAA.P11 12-DEC-80 15:59

LOAD DEVICE PROTECTION TABLE

.SBTTL LOAD DEVICE PROTECTION TABLE

```

:////////////////////
:/ THIS TABLE IDENTIFIES THE LOAD DEVICE TO THE SUPERVISOR, SO THAT IT CAN BE
:/ PROTECTED FROM TESTING. IF DESIRED.
:////////////////////

```

```

3495
3496
3497
3498
3499
3500
3501
3502 017614
3503 017614
3504 017614 177777
3505 017616 177777
3506 017620 177777
3507 017622

```

BGNPROT

```

.WORD -1 ;DON'T CHK CSR ADRS
.WORD -1 ;DON'T CHK MASSBUS UNIT NO.
.WORD -1 ;DON'T CHK DRIVE NO.
ENDPROT

```

L\$PROT::



CVDMAA.P11 12-DEC-80 15:59

INITIALIZE SECTION

.SBTTL INITIALIZE SECTION

:/ THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED  
:/ AT THE BEGINNING OF THE TEST SEQUENCE ON THE NEXT UNIT.

3508  
3509  
3510  
3511  
3512  
3513  
3514  
3515 017622  
3516 017622  
3517  
3518 017622 010637 002324  
3519  
3520 017626  
3521 017626 012700 000040  
3522 017632 104447  
3523 017634  
3524 017634 103417  
3525  
3526 017636  
3527 017636 012700 000037  
3528 017642 104447  
3529 017644  
3530 017644 103435  
3531  
3532 017646  
3533 017646 012700 000035  
3534 017652 104447  
3535 017654  
3536 017654 103433  
3537  
3538 017656  
3539 017656 012700 000036  
3540 017662 104447  
3541 017664  
3542 017664 103401  
3543 017666 000436  
3544  
3545 017670 000137 020074  
3546  
3547 017674  
3548  
3549  
3550  
3551 017674 005037 002346  
3552 017700  
3553 017700 012746 000000  
3554 017704 012746 020174  
3555 017710 012746 000004  
3556 017714 012746 000003  
3557 017720 104437  
3558 017722 062706 000010  
3559 017726 005737 177564  
3560 017732  
3561 017732 012700 000004  
3562 017736 104436  
3563

```
BGNINIT
LSINIT::
MOV SP,PSTACK ;SAVE BASE-LEVEL STACK POINTER
;SEE IF PROGRAM JUST STARTED, BR IF YES
READF #EF.START
BCOMPLETE STARST
;SEE IF PROGRAM JUST RESTARTED, BR IF YES
READF #EF.RESTART
BCOMPLETE RESTRT
;SEE IF THIS IS A NEW PASS, BR IF YES
READF #EF.NEW
BCOMPLETE NEWST
;SEE IF PROGRAM WAS JUST CONTINUED
READF #EF.CONTINUE
BCOMPLETE 10$
BR GETPRM
10$: JMP CONTIN ;(THIS IS TO FAR AWAY FOR A 'BR' INSTRUCTION)
STARST: ;ENTER HERE IF "START" COMMAND ISSUED
; TEST FOR THE PRESENCE OR ABSENCE OF A CONSOLE TERMINAL.
CLR CONSOL ;RESET THE CONSOLE TERMINAL FLAG
SETVEC #4,#CONST,#0 ;SETUP BUS TIMEOUT VECTER TO TEST FOR A CONSOLE
MOV #0,-(SP)
MOV #CONST,-(SP)
MOV #4,-(SP)
MOV #3,-(SP)
TRAP CSSVEC
ADD #10,SP
TST @#177564 ;TRY TO ACCESS THE CONSOLE TERMINAL'S 'XCSR'
CLRVEC #4 ;WE SHOULD BE THROUGH WITH THIS BY NOW
MOV #4,RO
TRAP CSCVEC
```

CVDMAA.P11 12-DEC-80 15:59

## INITIALIZE SECTION

```

3564 017740
3565
3566
3567 017740 005037 002342
3568
3569 017744
3570
3571 017744 012737 177777 002322
3572 017752 005237 002340
3573 017756 012737 000001 002344
3574
3575
3576 017764
3577 017764 005237 002322
3578 017770
3579 017770 013700 002322
3580 017774 104442
3581 017776 010001
3582 020000
3583 020000 103403
3584 020002 006337 002344
3585 020006 000766
3586
3587 020010 053737 002344 002342 10$:
3588 020016 006337 002344
3589
3590
3591
3592 020022 012100
3593 020024 012703 000020
3594 020030 012702 002352
3595 020034 010022
3596 020036 005200
3597 020040 077303
3598
3599
3600
3601 020042 012100
3602 020044 010037 002412
3603 020050 022020
3604 020052 010037 002414
3605
3606 020056 012100
3607 020060 006200
3608 020062 006200
3609 020064 006200
3610 020066 006200
3611 020070 010037 002416
3612
3613 020074
3614
3615 020074
3616 020074 013746 002416
3617 020100 012746 005152
3618 020104 013746 002412
3619 020110 012746 000003

```

```

RESTRT:          ;ENTER HERE IF 'RESTART' COMMAND ISSUED

;CLEAR DEVICE MAP
CLR      DEVMAP

NEWST:          ;ENTER HERE BEFORE EACH TEST

MOV      #-1,LOGDEV      ;RESET LOGICAL DEVICE TO -1
INC      FRSPAS          ;INCREMENT NO. OF PASSES AFTER LOAD
MOV      #BIT0,DEVPTR    ;INIT DEVICE MAP BIT POINTER
; GET UNIBUS ADDRESS, VECTOR, PRIORITY LEVEL, SWITCH PACKS, TEST
; CONNECTOR INFORMATION FOR THIS LOGICAL DEVICE
GETPRM:
INC      LOGDEV          ;INCREMENT LOGICAL DEVICE NUMBER
GPHARD  LOGDEV,R1       ;GET P-TABLE POINTER INTO R1
MOV      LOGDEV,R0
TRAP    CS$GPHRD
MOV      R0,R1

BCOMplete 10$         ;BR IF DEVICE AVAILABLE
MOV      LOGDEV,R0
TRAP    CS$GPHRD
MOV      R0,R1

ASL      DEVPTR          ;IF UN-AVAILABLE, SHIFT DEVICE MAP BIT POINTER
BR       GETPRM        ; AND SKIP THIS DEVICE

10$:  BIS      DEVPTR,DEVMAP ;ELSE, SET BIT FOR THIS DEVICE IN DEVICE MAP
ASL      DEVPTR          ;SHIFT DEVICE MAP BIT POINTER

; 'R1' WAS RETURNED WITH A POINTER TO THE CURRENT 'P-TABLE'

MOV      (R1)+,R0       ;GET THE DEVICE CSR ADDRESS
MOV      #16,R3         ;WE HAVE TO SETUP THIS MANY ADDRESS POINTERS
MOV      #MPCSR,R2      ;THIS IS THE ADDRESS OF THE FIRST POINTER
12$:  MOV      R0,(R2)+   ;SETUP ONE CSR POINTER
INC      R0              ;POINT TO THE NEXT CSR ADDRESS
SOB     R3,12$         ;LOOP AS LONG AS THERE ARE MORE TABLE ENTRIES
;ELSE, FALL THROUGH TO CONTINUE GETTING MORE
; P-TABLE DATA

MOV      (R1)+,R0       ;GET INTERRUPT VECTOR
MOV      R0,MPIVEC      ;SETUP 'A' VECTOR POINTER
CMP      (R0)+,(R0)+    ;ADD 4 TO VECTOR TO GET ADDRESS OF 'B' VECTOR
MOV      R0,MPOVEC      ;SETUP 'B' VECTOR POINTER

MOV      (R1)+,R0       ;GET DMV11 DEVICE PRIORITY
ASR      R0              ; RE-POSITION IT
ASR      R0
ASR      R0
ASR      R0
MOV      R0,MPRIOR     ;SETUP OUR VARIABLE FOR INT. VECTOR INIT'S

CONTIN:         ;ENTER HERE WHEN A 'CONTINUE' COMMAND IS ISSUED

SETVEC @MPIVEC,@MPIHAN,@MPRIOR ;SETUP 'A' INT. VECTOR
MOV      @MPRIOR,-(SP)
MOV      @MPIHAN,-(SP)
MOV      @MPIVEC,-(SP)
MOV      #3,-(SP)

```



CVDMAA.P11 12-DEC-80 15:59

## INITIALIZE SECTION

```

3620 020114 104437
3621 020116 062706 000010
3622 020122 005037 005222
3623 020126
3624 020126 013746 002416
3625 020132 012746 005224
3626 020136 013746 002414
3627 020142 012746 000003
3628 020146 104437
3629 020150 062706 000010
3630 020154 005037 005274
3631 020160 005037 002330
3632
3633 020164 012737 000001 002336
3634 020172
3635 020172
3636 020172 104411
3637
3638
3639
3640
3641
3642 020174 012737 177777 002346
3643 020202 000002
3644

          TRAP      CSSVEC
          ADD      #10,SP
          CLR      IHILNK          ;WE DON'T WANT THE HANDLER TO LINK ELSEWHERE
          SETVEC   @AMPPOVEC,@MPOHAN,@MPRIOR ;SETUP 'B' INT. VECTOR
          MOV      @MPRIOR,-(SP)
          MOV      @MPOHAN,-(SP)
          MOV      @MPOVEC,-(SP)
          MOV      #3,-(SP)
          TRAP     CSSVEC
          ADD      #10,SP
          CLR      IHOLNK          ;WE DON'T WANT THE HANDLER TO LINK ELSEWHERE
          CLR      INTWCH         ;RESET 'INTERRUPT WATCH' FLAGS (BOTH 'A' & 'B')
          MOV      #1,FRSTIM       ;MARK FLAG FOR NEXT TIME THROUGH
          ENDINIT ;END OF 'INIT' CODE
          L10017:
          TRAP     CSINIT

; ***** SUBROUTINES USED BY 'INIT' CODE *****
;
; INTERRUPT HANDLER FOR CONSOLE TERMINAL PRESENCE TESTING
CONTST: MOV      #-1,CONSOL       ;INDICATE THAT NO CONSOLE TERMINAL EXISTS!
          RTI                    ;RETURN

```

CVDMAA.P11 12-DEC-80 15:59

AUTO DROP UNIT SECTION

.SBTTL AUTO DROP UNIT SECTION

```

://////
: THE AUTO DROP CODING DETERMINES WHETHER OR NOT THE DEVICE WHOSE P-TABLE
: WAS JUST OBTAINED IS READY FOR TESTING, AND IT IS DROPPED IF NOT READY.
://////

```

```

:*****

```

```

THIS ALGORITHM IS THE SAME AS TEST # 1 EXCEPT THAT TEST 1
WILL JUST REPORT THE FAILURE AND GO ON -- THIS ROUTINE WILL CAUSE THE
DEVICE TO BE DROPPED IF A BUS-TIMEOUT OCCURS WHEN ANY OF THE CSR'S
ARE ACCESSED WITH EITHER A 'TST' OR 'TSTB' INSTRUCTION.

```

```

:-----*****

```

```

3645
3646
3647
3648
3649
3650
3651
3652
3653
3654
3655
3656
3657
3658
3659
3660
3661 020204
3662 020204
3663 020204
3664 020204 012746 000000
3665 020210 012746 020322
3666 020214 012746 000004
3667 020220 012746 000003
3668 020224 104437
3669 020226 062706 000010
3670 020232 005037 002440
3671 020236 012702 000001
3672 020242 013703 002352
3673
3674 020246 105723
3675 020250 006302
3676 020252 103375
3677
3678 020254 013703 002352
3679 020260 012702 000001
3680 020264 005723
3681 020266 006302
3682 020270 006302
3683 020272 103374
3684
3685 020274
3686 020274 012700 000004
3687 020300 104436
3688 020302 005737 002440
3689 020306 001403
3690 020310
3691 020310 013700 002322
3692 020314 104451
3693 020316 000240
3694 020320
3695 020320
3696 020320 104461
3697 020322 050237 002440
3698 020326 000002

```

```

BGNAUTO
SETVEC #4,#AD.HIT,#0 ;SETUP INVALID-ADDRESS TRAP VECTOR
LSAUTO::
MOV #0,-(SP)
MOV #AD.HIT,-(SP)
MOV #4,-(SP)
MOV #3,-(SP)
TRAP C$SVEC
ADD #10,SP

CLR TMP0 ;INITIALIZE TRAP FLAG REGISTER
MOV #1,R2 ;FLAG BIT
MOV BSEL0,R3 ;INIT ADDRESS POINTER

1$: TSTB (R3)+ ;ACCESS THE CSR'S BY BYTES.
ASL R2
BCC 1$

MOV BSEL0,R3 ;RE-INIT ADDRESS POINTER
MOV #1,R2 ;RE-INIT FLAG BIT
2$: TST (R3)+ ;ACCESS THE CSR'S BY WORDS.
ASL R2
ASL R2
BCC 2$

CLRVEC #4 ;RESTORE THE VECTOR TO DS

MOV #4,R0
TRAP C$CVEC
TST TMP0 ;DID WE GET HIT WITH AN INVALID ADDRESS TRAP?
BEQ AD.OK ;NO, EXIT TEST
DODU LOGDEV ;YES, DROP THIS LOGICAL DEV.

AD.OK: NOP ;(FOR PATCHING IN A HALT IF NECESSARY)
ENDAUTO

L10020:
AD.HIT: BIS R2,TMP0 ;FLAG THE HIT IF WE GET IT!
RTI ;RETURN
TRAP C$AUTO

```



CVDMAA.P11 12-DEC-80 15:59

CLEANUP CODING SECTION

.SBTTL CLEANUP CODING SECTION

```

:////////////////////
:// THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED
:// AT THE END OF THE TEST SEQUENCE ON A PARTICULAR UNIT.
:////////////////////

```

```

3699
3700
3701
3702
3703
3704
3705
3706 020330
3707 020330
3708 020330
3709 020330 013700 002412
3710 020334 104436
3711 020336
3712 020336 013700 002414
3713 020342 104436
3714 020344
3715 020344
3716 020344 104412

```

BGNCLN

CLRVEC @MPIVEC

CLRVEC @MPOVEC

ENDCLN

```

LSCLEAN::
;RETURN VECTORS TO SUPERVISOR

```

```

MOV @MPIVEC,RO
TRAP CSCVEC

```

```

MOV @MPOVEC,RO
TRAP CSCVEC

```

```

L10021:
TRAP CSCLEAN

```

CVDMAA.P11 12-DEC-80 15:59

DROP UNIT SECTION

3717  
3718  
3719  
3720  
3721  
3722  
3723  
3724 020346  
3725 020346  
3726  
3727 020346 104433  
3728 020346 104433  
3729 020350  
3730 020350  
3731 020350 104453

.SBTTL DROP UNIT SECTION

```

://////
:// THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
:// TO NO LONGER BE TESTED.
://////

```

BGNDU

```

:ISSUE UNIBUS RESET TO CLEAN UP
  BRESET

```

ENDDU

LSDU::

TRAP CSRESET

L10022:

TRAP CSDU



CVDMAA.P11 12-DEC-80 15:59

ADD UNIT SECTION

3732  
3733  
3734  
3735  
3736  
3737  
3738  
3739  
3740  
3741  
3742  
3743  
3744

020352  
020352  
020352  
020352  
020352 104452

.SBTTL ADD UNIT SECTION

:/ THE ADD-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE  
:/ TO BE (A) TESTED FOR THE FIRST TIME, OR (B) RESUMED IN TESTING. IF  
:/ 'EF.AUNIT' IS SET, THE UNIT WILL BE TESTED AS A NEW UNIT.

BGNAU  
ENDAU

LSAU::  
L10023: TRAP CSAU

CVDMAA.P11 12-DEC-80 15:59

TEST 1 -- DMV-11 AVAILABILITY

.SBTTL TEST 1 -- DMV-11 AVAILABILITY

```

:*****
:*
:*      TEST 1 -- DMV-11 AVAILABILITY
:*
:* EACH NORMALLY USED CSR IS ACCESSED WITH A 'TST' OR 'TSTB' INSTRUCTION AND IF
:* A BUS TIMEOUT OCCURS (INTERRUPT @ VECTOR ADDR 4) A FLAG WILL BE SET SHOWING
:* WHICH CSR ADDR AND INSTRUCTION FAILED. 'T1.HSW' REFLECTS 'TST' INSTRUCTIONS
:* AND 'T1.HSB' REFLECTS 'TSTB' INSTRUCTIONS.
:*
:*      EXAMPLES:
:*
:*      IF 'TSTB @BSEL1' FAILS, BIT # 1 OF 'T1.HSB' WILL BE SET.
:*      IF 'TST @SEL4' FAILS, BIT # 4 OF 'T1.HSW' WILL BE SET
:*      (NOTE: ONLY EVEN BITS IN 'T1.HSW' CAN BE SET).
:*
:*      THE FLAG WORDS ARE OUTPUT IN BINARY AS 'EXTENDED ERROR INFORMATION'.

```

\*\*\*\*\*

3745  
3746  
3747  
3748  
3749  
3750  
3751  
3752  
3753  
3754  
3755  
3756  
3757  
3758  
3759  
3760  
3761  
3762  
3763  
3764  
3765  
3766  
3767  
3768  
3769  
3770  
3771  
3772  
3773  
3774  
3775  
3776  
3777  
3778  
3779  
3780  
3781  
3782  
3783  
3784  
3785  
3786  
3787  
3788  
3789  
3790  
3791  
3792  
3793  
3794  
3795  
3796  
3797  
3798  
3799  
3800

```

020354
020354 005037 020520
020360 012702 000001
020364 013703 002352
020370
020370 012746 000000
020374 012746 020512
020400 012746 000004
020404 012746 000003
020410 104437
020412 062706 000010
020416 105723
020420 006302
020422 103375
020424 013737 020520 020522
020432 005037 020520
020436 012702 000001
020442 013703 002352
020446 005723
020450 006302
020452 006302
020454 103374
020456
020456 012700 000004
020462 104436
020464 005737 020520
020470 001003
020472 005737 020522
020476 001404
020500

```

```

:      BGNTST
:
:      T1::
:      CLR      T1.HSW      ;INITIALIZE TRAP FLAG REGISTER
:      MOV      #1,R2      ;FLAG BIT FOR BYTE ACCESSED CSR 0.
:      MOV      BSEL0,R3   ;INIT ADDRESS POINTER
:      SETVEC   #4,#T1.HIT,#0 ;SETUP INVALID-ADDRESS TRAP VECTOR
:
:      MOV      #0,-(SP)
:      MOV      #T1.HIT,-(SP)
:      MOV      #4,-(SP)
:      MOV      #3,-(SP)
:      TRAP    CS$VEC
:      ADD     #10,SP
:
1$:      TSTB    (R3)+      ;ACCESS THE CSR'S BY BYTES.
:      ASL     R2
:      BCC     1$
:
:      MOV     T1.HSW,T1.HSB ;MOVE BYTE INTERRUPT FLAG TO PROPER LOCATION.
:      CLR     T1.HSW      ;INITIALIZE TRAP FLAG REGISTER
:      MOV     #1,R2      ;FLAG BIT FOR WORD ACCESSED CSR 0.
:      MOV     BSEL0,R3   ;RE-INIT ADDRESS POINTER
:
2$:      TST     (R3)+      ;ACCESS THE CSR'S BY WORDS.
:      ASL     R2
:      ASL     R2
:      BCC     2$
:
:      CLRVEC  #4          ;RESTORE THE VECTOR TO DS
:
:      TST     T1.HSW      ;DID WE GET AN INVALID ADDRESS TRAP?
:      BNE     3$          ;YES, REPORT FAILURE
:      TST     T1.HSB
:      BEQ    T1.OK
:      GEDF   T1.EHD,T1.EM1 ;YES, REPORT THE ERROR
:
:      MOV     #4,R0
:      TRAP   CS$VEC

```



CVDMAA.P11 12-DEC-80 15:59

TEST 1 -- DMV-11 AVAILABILITY

```

3801                                     ; 'DEVICE FATAL' ERROR # 9
3802 020500 104455                       TRAP      C$ERDF
3803 020502 000011                       .WORD    9
3804 020504 020642                       .WORD    T1.EHD
3805 020506 020524                       .WORD    T1.EM1
3806 020510                               T1.OK:   ENDTST
3807 020510                               L10024:
3808 020510 104401                       TRAP      C$ETST
3809
3810 020512 050237 020520                T1.HIT:  BIS      R2,T1.HSW      ;FLAG THE HIT IF WE GET IT!
3811 020516 000002                       RTI      ;RETURN
3812
3813 020520 000000                       T1.HSW:  .WORD    0              ;INVALID ADDRESS TRAP FLAG WORD:
3814                                     ;BITS SET INDICATE TRAPS ON WORD ACCESSES
3815                                     ;(BIT # SET = CSR # THAT FAILED)
3816 020522 000000                       T1.HSB:  .WORD    0              ;INVALID ADDRESS TRAP FLAG WORD:
3817                                     ;BITS SET INDICATE TRAPS ON BYTE ACCESSES
3818                                     ;(BIT # SET = CSR # THAT FAILED)
3819 020524                               BGNMSG   T1.EM1
3820 020524                               PRINTB   #T1.1,MPCSR            ;IDENTIFY ERROR AND ON WHAT DEVICE
3821 020524                               T1.EM1::
3822 020524 013746 002352                   MOV      MPCSR,-(SP)
3823 020530 012746 020671                   MOV      #T1.1,-(SP)
3824 020534 012746 000002                   MOV      #2,-(SP)
3825 020540 010600                           MOV      SP,R0
3826 020542 104414                           TRAP     C$PNTB
3827 020544 062706 000006                   ADD      #6,SP
3828 020550                               PRINTX   #T1.2                ;IF REQUESTED, ALSO INDICATE MISSES (TRAPS)
3829 020550 012746 020753                   MOV      #T1.2,-(SP)
3830 020554 012746 000001                   MOV      #1,-(SP)
3831 020560 010600                           MOV      SP,R0
3832 020562 104415                           TRAP     C$PNTX
3833 020564 062706 000004                   ADD      #4,SP
3834 020570                               PRINTX   #T1.3
3835 020570 012746 021006                   MOV      #T1.3,-(SP)
3836 020574 012746 000001                   MOV      #1,-(SP)
3837 020600 010600                           MOV      SP,R0
3838 020602 104415                           TRAP     C$PNTX
3839 020604 062706 000004                   ADD      #4,SP
3840 020610                               PRINTX   #T1.4,T1.HSW,T1.HSB
3841 020610 013746 020522                   MOV      T1.HSB,-(SP)
3842 020614 013746 020520                   MOV      T1.HSW,-(SP)
3843 020620 012746 021060                   MOV      #T1.4,-(SP)
3844 020624 012746 000003                   MOV      #3,-(SP)
3845 020630 010600                           MOV      SP,R0
3846 020632 104415                           TRAP     C$PNTX
3847 020634 062706 000010                   ADD      #10,SP
3848 020640                               ENDMSG
3849 020640                               L10025:
3850 020640 104423                       TRAP      C$MSG
3851
3852                                     .NLIST  BEX
020642 053101 044501 040514 T1.EHD: .ASCIZ 'AVAILABILITY TEST (#1)'
020671 045 022516 042101 T1.1: .ASCIZ '%XZADMV-11 @ %XA NOT RESPONDING TO CSR ACCESSING'
020753 045 031116 051445 T1.2: .ASCIZ '%N2XS21XASEL #XS11XABSEL #'
021006 047045 051445 032461 T1.3: .ASCIZ '%XZS15XAE C A 8 6 4 2 0 FEDCBA9876543210'

```

CVDMAA.P11 12-DEC-80 15:59

TEST 1 -- DMV-11 AVAILABILITY

021060 047045 022462 020101

T1.4: .ASCIZ '%N2%A TRAP FLAGS:%B16%S2%B16'

3853 021120

.LIST  
.EVEN

BEX



CVDMAA.P11 12-DEC-80 15:59

TEST 2 -- MASTER CLEAR, RUN MICRODIAGNOSTICS

.SBTTL TEST 2 -- MASTER CLEAR, RUN MICRODIAGNOSTICS

3854  
3855  
3856  
3857  
3858  
3859  
3860  
3861  
3862  
3863  
3864  
3865  
3866  
3867  
3868  
3869  
3870  
3871  
3872  
3873  
3874  
3875  
3876  
3877  
3878  
3879  
3880  
3881  
3882  
3883  
3884  
3885  
3886  
3887  
3888  
3889  
3890  
3891  
3892  
3893  
3894  
3895  
3896  
3897  
3898  
3899  
3900  
3901  
3902  
3903  
3904  
3905  
3906  
3907  
3908  
3909

021120

021120 004737 003614  
021124 103002  
021126 104460  
021130 000436  
  
021132 005001  
021134 005002  
021136 016203 003040  
021142 062702 000002  
021146 126271 003040 002352  
021154 001005  
021156 005202  
021160 005201  
021162 005201  
  
021164 077310  
021166 000417  
  
021170 017137 002352 002312  
021176 004737 004434  
021202 016237 003040 002310  
021210 006201  
021212 010137 002334  
  
021216 104455  
021220 000012  
021222 014667

```

:*****
:*
:*      TEST 2 -- MASTER CLEAR, RUN MICRODIAGNOSTICS
:*
:*  A MASTER CLEAR IS ISSUED TO THE DMV-11, AND THE PROGRAM ALLOWS SUFFICIENT
:*  TIME FOR THE MICRODIAGNOSTICS TO BE PERFORMED. THESE MICRODIAGNOSTICS RESIDE
:*  IN 6502 PROGRAM MEMORY, AND THOROUGHLY VERIFY THE OPERATION OF THE 6502
:*  MICROPROCESSOR CHIP. THEN, THEY CHECK OUT THE DATA RAM, THE 6502'S ACCESS TO
:*  THE CSR'S, AND PERFORM A SIMPLE MESSAGE TEST USING THE 6522 CHIP AND THE
:*  USYRT, WITH INTERNAL LOOPBACK.
:*
:*  NEXT, THE LSI-11 PROGRAM READS THE THE CSR'S (SEL0-SEL6) AND CHECKS THEM FOR
:*  THEIR EXPECTED INITIALIZED STATES. IF AN ERROR HAS OCCURRED IN THE MICRO-
:*  DIAGNOSTICS THE NUMBER OF THE FAILING TEST WILL BE FOUND IN SEL4, AND RUN
:*  (BIT 7) WILL NOT BE SET IN BSEL1.
:*****
:
:      BGNTST
:
:      T2::
:  ISSUE A MASTER CLEAR, AND DELAY FOR MICRO-DIAGNOSTICS TO COMPLETE BY CALLING
:  SUBROUTINE MASCLR.
:
:      JSR    PC,MASCLR      ; -ATTEMPT- TO RUN THE MICRO-DIAGNOSTIC
:      BCC    8$            ; IF NO ERROR, PROCEED
:      ERROR  ; ELSE, REPORT IT AND
:
:      BR     24$           ; EXIT THIS TEST
:
:      TRAP   CSERROR
:
:  ; FIRST, INITIALIZE INDEX REGISTERS
:  8$: CLR    R1            ; R1 IS THE INDEX OF THE BYTE SELECT TABLE
:      CLR    R2            ; R2 IS THE INDEX OF THE RESULTS TABLE
:      MOV    RESFMC(R2),R3 ; GET THE NUMBER OF PATTERNS IN RESULTS TABLE
:      ADD    #2,R2         ; MOVE POINTER TO NEXT BYTE
:  2$: CMPB   RESFMC(R2),@BSEL(R1) ; COMPARE EXPECTED RESULTS WITH CSR'S.
:      BNE    1$            ; A MISMATCH IS A DEVICE FATAL ERROR
:      INC    R2            ; INCREMENT TABLE POINTER
:      INC    R1            ; INCREMENT POINTER
:      INC    R1            ; BY 2 (WORD INCREMENT)
:
:      SOB   R3,2$         ; CONTINUE TO LOOP THROUGH TABLE
:
:      BR    24$           ; TEST COMPLETE WITH NO ERRORS, GO END TEST.
:
:  1$: MOV    @BSEL(R1),BDATA ; GET DATA WORD THAT FAILED
:      JSR    PC,GETBSR     ; GET THE BSEL REGISTERS FOR DUMPING
:      MOV    RESFMC(R2),GDATA ; GET EXPECTED RESULT FROM TABLE
:      ASR   R1            ; CONVERT WORD OFFSET TO BYTE CSR #
:      MOV    R1,REGNUM
:      GEDF   EM14,ERR2    ; DEVICE FATAL ERROR, REPORT IT AND END TEST
:
:      TRAP   CSERDF
:      .WORD 10
:      .WORD EM14

```

CVDMAA.P11 12-DEC-80 15:59

TEST 2 -- MASTER CLEAR, RUN MICRODIAGNOSTICS

3910 021224 005304  
3911 021226  
3912 021226  
3913 021226 104401

24\$: ENDTST

.WORD ERR2  
L10026: TRAP C\$ETST



CVDMAA.P11 12-DEC-80 15:59

TEST 3 -- CSR ADDRESSING

.SBTTL TEST 3 -- CSR ADDRESSING

3914  
3915  
3916  
3917  
3918  
3919  
3920  
3921  
3922  
3923  
3924  
3925  
3926  
3927  
3928  
3929  
3930  
3931  
3932  
3933  
3934  
3935  
3936  
3937  
3938  
3939  
3940  
3941  
3942  
3943  
3944  
3945  
3946  
3947  
3948  
3949  
3950  
3951  
3952  
3953  
3954  
3955  
3956  
3957  
3958  
3959  
3960  
3961  
3962  
3963  
3964  
3965  
3966  
3967  
3968  
3969

021230

021230 012703 000010  
021234 013701 002352  
021240 012702 003062  
021244 005021  
021246 005022  
021250 005022  
021252 077304  
  
021254 005002  
021256 012703 000020  
021262 105772 002352  
021266 001035  
021270 005722  
021272 077305  
  
021274 005001

\*\*\*\*\*  
\*  
\* TEST 3 -- CSR ADDRESSING  
\*  
\* FIRST, HALT THE 6502 UP BY CLEARING ALL CSRS. THEN, WRITE A DIFFERENT WORD  
\* OF DATA PATTERN A INTO EACH OF BSEL0-17, AND AFTER EACH WRITE, READ AND  
\* COMPARE ALL REGS TO EXPECTED VALUES.  
\*  
\* DATA PATTERN A = 001, 002, 004, 010, 020, 040, 100, 200, 052, 300, 140,  
\* 060, 030, 014, 006, 003  
\*  
\*\*\*\*\*

BGNTST

T3::

\*\*\*\*\* DETAILED TEST DESCRIPTION \*\*\*\*\*  
THIS TEST PROCEEDS AS FOLLOWS:

- (1) CLEAR ALL CSRS AND VERIFY SAME (CLEARING BSEL01 HALTS 6502)
  - (2) WRITE 01 INTO BSEL0; VERIFY BSEL0=01, ALL OTHERS=0
  - (3) WRITE 02 INTO BSEL1; VERIFY BSEL0=01, BSEL1=02, ALL OTHERS=0.
  - (4) WRITE 04 INTO BSEL2; VERIFY BSEL0=01, BSEL1=02, BSEL2=04, ALL OTHERS=0
  - (5) => (17) CONTINUE TO INCREMENTALLY WRITE DATA-PATTERN-A INTO THE BSR'S, CHECKING ALL BSR'S AFTER EACH WRITE, UNTIL BSR'S COMPLETELY FILLED WITH DATA-PATTERN-A.
- NOTE: IF AN ERROR OCCURS, THE FIRST BAD BSR NUMBER AND GOOD/BAD VALUES ARE GIVEN, FOLLOWED BY A COMPLETE BSR DUMP.

: CLEAR DMV CSRS AND RESULTS TABLE

```
MOV #10,R3 ;GET # OF CSRS
MOV @BSEL,R1 ;GET 1ST CSR ADDRESS
MOV #RESFT3,R2 ;GET 1ST RESULTS TABLE ADDRESS
1$: CLR (R1)+ ;CLEAR CSR, BUMP POINTER
CLR (R2)+ ;CLEAR RESULTS TABLE LOC., BUMP POINTER
CLR (R2)+ ; AND DO AGAIN
SOB R3,1$ ;LOOP UNTIL ALL DONE
```

: NOW VERIFY CSRS ARE ALL ZEROED

```
CLR R2 ;CLEAR BSR ADDRESS INDEX
MOV #CSREGS,R3 ;GET # OF CSRS
2$: TSTB @BSEL(R2) ;IS THIS CSR=0 ?
BNE 5$ ;IF NO: GO REPORT ERROR
TST (R2)+ ; YES: BUMP INDEX
SOB R3,2$ ;DO UNTIL ALL BSRS CHECKED
```

\*\*\*\*\*  
: INITIALIZE INDEX REGISTERS

```
CLR R1 ;INITIALIZE PATTERN INDEX REGISTER
```

CVDMAA.P11 12-DEC-80 15:59

## TEST 3 -- CSR ADDRESSING

```

3970 021276 012703 000020      MOV    #CSREGS,R3      ;GET NUMBER OF CSR'S
3971
3972      ; THE FIRST WORD OF THE DATA TABLE CONTAINS THE NUMBER OF PATTERNS IN
3973      ; THE TABLE:
3974 021302 016104 002504      MOV    PATA(R1),R4     ;INITIALIZE NUMBER OF PATTERNS COUNT
3975 021306 005721              TST    (R1)+          ;MOVE TABLE POINTER
3976
3977      ; PUT NEXT PATTERN OF DATA INTO NEXT REGISTER AND TEST AREA:
3978
3979      ; CALCULATE INDEX INTO DATA AREA AND TO REGISTER
3980
3981 021310 010102              3$:    MOV    R1,R2      ;GET INDEX INTO TEST DATA AREA
3982 021312 005742              TST    -(R2)         ;IT'S ONE WORD TOO LARGE
3983 021314 006302              ASL    R2            ;CONVERT FROM BYTE TO WORD INDEX
3984
3985      ; NOW, SETUP THE EXPECTED RESULTS AREA AND LOAD THE SELECT REGISTER
3986
3987 021316 116162 002504 003062  MOVB   PATA(R1),RESFT3(R2) ;UPDATE THE EXPECTED RESULTS TABLE
3988 021324 116172 002504 002352  MOVB   PATA(R1),@BSEL(R2) ;PUT PATTERN INTO THE CSR
3989
3990 021332 005201              INC    R1            ;BUMP DATA POINTER FOR NEXT TIME AROUND
3991 021334 005002              CLR    R2            ;INITIALIZE TABLE INDEX
3992 021336 012703 000020      MOV    #CSREGS,R3     ;INITIALIZE NUMBER OF REGISTERS
3993
3994 021342 126272 003062 002352  4$:    CMPB   RESFT3(R2),@BSEL(R2) ;COMPARE CSR WITH RESULTS TABLE
3995 021350 001004              BNE    5$           ;A MISMATCH IS A DEVICE FATAL ERROR
3996 021352 005722              TST    (R2)+        ;BUMP TABLE POINTER BY 2 (WORD INCREMENT)
3997 021354 077306              SOB    R3,4$       ;CONTINUE TO READ & MATCH ALL REGISTERS BEFORE
3998                          ;LOADING THE NEXT PATTERN INTO NEXT REGISTER
3999
4000                          SOB    R4,3$       ;LOOP UNTIL ALL PATTERNS ARE TESTED
4001 021360 000417              BR     24$         ;TEST COMPLETE **<< NO ERRORS >>**
4002
4003      ;--PREPARE THE FAILURE MESSAGE --
4004
4005 021362 016237 003062 002310  5$:    MOV    RESFT3(R2),GDATA ;GET THE EXPECTED RESULT FROM TABLE
4006 021370 017237 002352 002312  MOV    @BSEL(R2),BDATA ;GET THE FAILED WORD
4007 021376 004737 004434      JSR    PC,GETBSR     ;GET THE BSEL REGISTERS FOR DUMPING
4008 021402 006202              ASR    R2            ;CONVERT WORD OFFSET TO BYTE CSR ADDRESS
4009 021404 010237 002334      MOV    R2,REGNUM     ;GET THE REGISTER THAT FAILED
4010 021410              GEDF   EM6,ERR2     ;ERROR **** DEVICE FATAL ****
4011                          ;      'DEVICE FATAL' ERROR # 11
4012 021410 104455              TRAP   C$ERDF
4013 021412 000013              .WORD 11
4014 021414 014540              .WORD EM6
4015 021416 005304              .WORD ERR2
4016 021420              24$:    ENDTST
4017 021420              L10027:
4018 021420 104401              TRAP   C$ETST

```



CVDMAA.P11 12-DEC-80 15:59

## TEST 4 -- CSR REGISTERS DATA READ/WRITE

.SBTTL TEST 4 -- CSR REGISTERS DATA READ/WRITE

4019  
4020  
4021  
4022  
4023  
4024  
4025  
4026  
4027  
4028  
4029  
4030  
4031  
4032  
4033  
4034  
4035  
4036  
4037  
4038  
4039  
4040  
4041  
4042  
4043  
4044  
4045  
4046  
4047  
4048  
4049  
4050  
4051  
4052  
4053  
4054  
4055  
4056  
4057  
4058  
4059  
4060  
4061  
4062  
4063  
4064  
4065  
4066  
4067  
4068  
4069  
4070  
4071  
4072  
4073  
4074

021422  
021422 004737 003762  
  
021426 103002  
021430  
021430 104460  
021432 000453  
  
  
  
  
  
021434 005001  
021436 005002  
021440 016103 002526  
021444 062702 000002  
  
021450 113777 000101 160676  
021456 116137 002526 002310  
  
  
  
021464 022702 000002  
021470 001003  
  
021472 142737 000300 002310  
021500 113772 002310 002352

```

*****
*
* TEST 4 -- CSR REGISTERS DATA READ/WRITE
* WRITE, READ, AND COMPARE EACH BYTE OF DATA PATTERN B INTO REGISTER BSEL0.
* THEN, REPEAT THIS USING EACH OF THE REMAINING CSR'S, BSEL1-BSEL17. WHEN BSEL1
* IS BEING TESTED, THE PROGRAM ALWAYS SETS BIT 7 IN THE DATA PATTERN SO THAT
* RUN WILL NOT BE CLEARED, AND IT ALWAYS CLEARS BIT6 SO THAT MCLR WILL NOT BE
* SET.
*
* DATA PATTERN B = 125, 252, 000, 377, 001, 002, 004, 010, 020, 040, 100,
*                   200, 376, 375, 373, 367, 357, 337, 277, 177, 000
*****
:
:      BGNTST
:      JSR      PC,MSTCLR      ;CALL MAINTENANCE READY INITIALIZATION. IF
:                                     T4::
:                                     ;MSTCLR SHOULD FAIL BECAUSE THE MRDY FLAG DOES
:                                     ;NOT BECOME SET, A DEVICE FATAL ERROR WILL BE
:                                     ;REPORTED, AND MSTCLR WILL SET THE 'C' BIT
:                                     ;IF NO ERROR, PROCEED
:                                     ;ELSE, REPORT IT AND
:
:      BCC      8$
:      ERROR
:
:      BR       24$           ;      EXIT THIS TEST
:
:      TRAP     C$ERROR
:
: NOTE - THE FIRST BYTE LOCATION OF THE PATTERN B TABLE, USED IN THIS TEST,
:        CONTAINS THE NUMBER OF TEST PATTERNS OF PATTERN B TABLE, NOT A
:        TEST PATTERN.
:
: FIRST, INITIALIZE INDEX AND COUNT REGISTERS
8$:   CLR      R1           ;R1 IS THE 'PATB' INDEX REGISTER
      CLR      R2           ;R2 IS THE CSR INDEX REGISTER
      MOV     PATB(R1),R3   ;R3 CONTAINS THE NUMBER OF BYTES IN PATB
      ADD     #2,R2        ;MOVE POINTER TO FIRST BYTE OF DATA
:
:      MOVB    101,@BSEL1   ;STOP THE MICRO-PROCESSOR!!!
1$:   MOVB    PATB(R1),GDATA ;GET THE PATB DATA BYTE, WE ARE TO USE
:
: DON'T GET CAUGHT BY THE NEXT INSTRUCTION! 'R2' IS AN OFFSET INTO A
: WORD TABLE WHICH CONTAINS THE ADDRESSES OF THE CSR'S. THEREFORE, WHEN
: R2 = 0 -- IT POINTS TO BSEL0'S ADDRESS, AND WHEN R2 = 2 -- IT POINTS TO
: BSEL1'S ADDRESS.
:
:      CMP     #2,R2       ;IS 'BSEL1' BEING TESTED?
:      BNE    2$          ;IF YES, ALTER PATB DATA SO THAT BIT 7 IS
:                          ;ALWAYS SET, AND BIT6 IS ALWAYS RESET.
:                          ;ELSE, USE PATB DATA AS IS.
:
:      BICB   #RUN!MCLR,GDATA ;FORCE PATTERN TO RESET BITS 7 & 6
2$:   MOVB    GDATA,@BSEL0(R2) ;PUT PATB DATA INTO REGISTER BEING TESTED

```

CVDMAA.P11 12-DEC-80 15:59

## TEST 4 -- CSR REGISTERS DATA READ/WRITE

```

4075 021506 123772 002310 002352      CMPB   GDATA,@BSELO(R2)  ; COMPARE PATTERN JUST WRITTEN
4076 021514 001414                    BEQ    5$                 ; TEST PASSES IF A MATCH. ELSE, DEVICE FATAL ERROR
4077                                     ;--PREPARE FOR THE FAILURE PRINTOUT--
4078
4079
4080 021516 010237 002334                    MOV    R2,REGNUM          ; GET THE REGISTER THAT FAILED
4081 021522 117237 002352 002312          MOVB  @BSELO(R2),BDATA    ; SCORE THE BAD DATA
4082 021530 004737 004434                    JSR   PC,GETBSR          ; GET THE BSEL REGISTERS FOR DUMPING
4083 021534                                     GEDF  EM7,ERR2           ; REPORT ERROR AND EXIT THE TEST
4084                                     ; 'DEVICE FATAL' ERROR # 12
4085 021534 104455                                     TRAP  C$ERDF
4086 021536 000014                                     .WORD 12
4087 021540 014564                                     .WORD EM7
4088 021542 005304                                     .WORD ERR2
4089 021544 000406
4090
4091 021546 005201                    5$:   INC    R1              ; MOVE TABLE POINTER
4092 021550 077336                    SOB   R3,1$              ; DECREMENT NUMBER OF PATTERNS LEFT. IF ZERO,EXIT.
4093                                     ; ELSE, CONTINUE TO PATTERN TEST REGISTER
4094 021552 005722                    TST   (R2)+              ; INCREMENT THE REGISTER INDEX BY 2
4095 021554 020227 000040          CMP   R2,#<CSREGS*2>    ; COMPARE REGISTER INDEX TO NUMBER OF CSR'S
4096 021560 101336                    BHI   1$                 ; IF R2 > 17, END THE TEST
4097
4098 021562                    24$:
4099 021562                    ENDTST
4100 021562
4101 021562 104401                    L10030: TRAP  C$ETST

```



CVDMAA.P11 12-DEC-80 15:59

TEST 5 -- BASIC MASTER CLEAR

.SBTTL TEST 5 -- BASIC MASTER CLEAR

4102  
4103  
4104  
4105  
4106  
4107  
4108  
4109  
4110  
4111  
4112  
4113  
4114  
4115  
4116  
4117  
4118  
4119  
4120  
4121  
4122  
4123  
4124  
4125  
4126  
4127  
4128  
4129  
4130  
4131  
4132  
4133  
4134  
4135  
4136  
4137  
4138  
4139  
4140  
4141  
4142  
4143  
4144  
4145  
4146  
4147  
4148  
4149  
4150  
4151  
4152  
4153  
4154  
4155  
4156  
4157

021564

021564 004737 003614

021570 103002

021572

021572 104460

021574 000441

021576 112777 000377 160546 8\$:

021604 122777 000377 160540

021612 001011

021614 004737 003614

021620 103002

021622

021622 104460

021624 000425

021626 122777 000000 160516 9\$:

021634 001421

021636 112737 000377 002310 2\$:

021644 117737 160502 002312

021652 004737 004434

021656 105077 160470

021662 012737 000000 002334

021670

021670 104455

021672 000015

021674 014515

021676 005304

021700 105077 160446 24\$:

\*\*\*\*\*  
\*  
\* TEST 5 -- BASIC MASTER CLEAR  
\*  
\* PERFORM AN INITIAL MASTER CLEAR. WRITE 377 INTO BSELO AND READ AND CHECK IT.  
\* THEN, ISSUE A MASTER CLEAR AND READ AND CHECK BSELO FOR 000.  
\*  
\*\*\*\*\*

BGNTST

T5::  
; ISSUE A MASTER CLEAR, AND DELAY FOR MICRO-DIAGNOSTICS TO COMPLETE BY CALLING  
; SUBROUTINE MASCLR.

JSR PC,MASCLR ; -ATTEMPT- TO RUN THE MICRO-DIAGNOSTIC  
; FAILURES WILL BE REPORTED BY THE SUBROUTINE  
; AS DEVICE FATAL AND THE 'C' BIT WILL BE SET  
BCC 8\$ ; IF NO ERROR, PROCEED  
ERROR ; ELSE, REPORT IT AND TRAP C\$ERROR  
BR 24\$ ; EXIT THIS TEST  
MOVB #377,@BSELO ; SET BSEL TO ALL ONES  
CMPB #377,@BSELO ; COMPARE  
BNE 2\$ ; A MISMATCH INDICATES A DEVICE FATAL ERROR

; ISSUE A MASTER CLEAR, AND DELAY FOR MICRO-DIAGNOSTICS TO COMPLETE BY CALLING  
; SUBROUTINE MASCLR.

JSR PC,MASCLR ; -ATTEMPT- TO RUN THE MICRO-DIAGNOSTIC  
; FAILURES WILL BE REPORTED BY THE SUBROUTINE  
; AS DEVICE FATAL AND THE 'C' BIT WILL BE SET  
BCC 9\$ ; IF NO ERROR, PROCEED  
ERROR ; ELSE, REPORT IT AND TRAP C\$ERROR  
BR 24\$ ; EXIT THIS TEST

CMPB #000,@BSELO ; THIS REGISTER SHOULD BE ZEROED DURING  
; INITIALIZATION  
BEQ 24\$ ; IF ZERO, \*\*\* TEST PASSES \*\*\*, ELSE REPORT ERROR

---PREPARE FOR THE FAILURE PRINTOUT---  
MOVB #377,GDATA ; ALL ONES IS EXPECTED DATA  
MOVB @BSELO,BDATA ; SOMETHING OTHER THAN ALL ONES WAS FOUND. SCORE IT.  
JSR PC,GETBSR ; GET THE BSEL REGISTERS FOR DUMPING  
CLRB @BSELO ; DISABLE INTERRUPTS AS A PRECAUTIONARY MEASURE  
MOV #0,REGNUM ; GET THE REGISTER THAT FAILED  
GEDF EMS,ERR2 ; REPORT DEVICE FATAL ERROR  
; 'DEVICE FATAL' ERROR # 13

TRAP C\$ERDF  
.WORD 13  
.WORD EMS  
.WORD ERR2  
; DISABLE INTERRUPTS AS A PRECAUTIONARY MEASURE

CVDMAA.P11 12-DEC-80 15:59

TEST 5 -- BASIC MASTER CLEAR

4158 021704  
4159 021704  
4160 021704 104401

ENDTST

L10031: TRAP CSETST





CVDMAA.P11 12-DEC-80 15:59

TEST 6 -- BUS RESET

```

4217 022026 104455
4218 022030 000016
4219 022032 014634
4220 022034 005304
4221 022036 000421
4222
4223 022040 117737 160306 002312 1$:  MOVB @BSELO,BDATA ;GET THE ACTUAL DATA
4224 022046 004737 004434          JSR PC,GETBSR ;GET THE BSEL REGISTERS FOR DUMPING
4225 022052 105077 160274          CLR8 @BSELO ;DISABLE INTERRUPTS AS A PRECAUTIONARY MEASURE
4226 022056 112737 000377 002310  MOVB #377,GDATA ;ALL ONES WAS EXPECTED DATA
4227 022064 012737 000000 002334  MOV #0,REGNUM ;GET THE REGISTER THAT FAILED
4228 022072          GEDF EM8,ERR2 ;BSELO COULD NOT BE SET TO ALL ONES
4229          ; 'DEVICE FATAL' ERROR # 15
4230 022072 104455
4231 022074 000017
4232 022076 014611
4233 022100 005304
4234 022102 105077 160244          24$: CLR8 @BSELO ;DISABLE INTERRUPTS AS A PRECAUTIONARY MEASURE
4235 022106          ENDTST
4236 022106
4237 022106 104401

```

```

TRAP CSERDF
.WORD 14
.WORD EM9
.WORD ERR2

```

```

TRAP CSERDF
.WORD 15
.WORD EM8
.WORD ERR2

```

```

L10032: TRAP CSETST

```



CVDMAA.P11 12-DEC-80 15:59

TEST 7 -- CSR, MAINTENANCE MICROCODE INTERACTION

.SBTTL TEST 7 -- CSR, MAINTENANCE MICROCODE INTERACTION

4238  
4239  
4240  
4241  
4242  
4243  
4244  
4245  
4246  
4247  
4248  
4249  
4250  
4251  
4252  
4253  
4254  
4255  
4256  
4257  
4258  
4259  
4260  
4261  
4262  
4263  
4264  
4265  
4266  
4267  
4268  
4269  
4270  
4271  
4272  
4273  
4274  
4275  
4276  
4277  
4278  
4279  
4280  
4281  
4282  
4283  
4284  
4285  
4286  
4287  
4288  
4289  
4290  
4291  
4292  
4293

```

:*****
:*
:* TEST 7 -- CSR, MAINTENANCE MICROCODE INTERACTION
:*
:* THIS TEST INVOKES THE MAINTENANCE REQUEST MECHANISM THROUGH WHICH THE LSI-11
:* AND 6502 CAN COMMUNICATE. FIRST, A MASTER CLEAR IS DONE WITH ONLY BIT 0
:* (MREQ) SET IN BSEL1. THE PROGRAM THEN CHECKS FOR THE SETTING OF BSEL2 BIT 7
:* (MRDY) BY THE MAINTENANCE MICROCODE WITHIN ABOUT 50 MICRO-SEC., AND IF MRDY
:* DOES NOT GET SET, AN ERROR IS REPORTED.
:*
:* NEXT, THE PROGRAM LOADS SEL4 WITH 000010 AND BSEL6 WITH 125. THEN, ALL CSR'S
:* ARE READ AND CHECKED FOR EXPECTED CONTENTS.
:*
:* BSEL2 IS THEN LOADED WITH A WRITE COMMAND, WHICH SHOULD CAUSE THE MICROCODE
:* TO TRANSFER THE 125 INTO BSEL0. ALL CSR'S ARE THEN READ AND CHECKED FOR
:* EXPECTED CONTENTS.
:*
:* THEN, THE PROGRAM LOADS 252 INTO BSEL0 AND READS AND CHECKS ALL CSR'S. BSEL2
:* IS THEN LOADED WITH A READ COMMAND, WHICH SHOULD CAUSE THE MICROCODE TO
:* TRANSFER THE 252 INTO BSEL6. ALL CSR'S ARE READ AND CHECKED.
:*
:*****

```

BGNTST

T7::

BGNSUB

T7.1:

```

JSR PC,MSTCLR
BCC 10$
ERROR

```

```

;PUT THE MICROPROCESSOR IN THE MAINTENANCE LOOP
;IF NO ERROR, PROCEED
;ELSE, REPORT IT AND

```

```

TRAP CSBSUB
TRAP CSERROR

```

JMP ENDT7

; EXIT THIS TEST

```

MOV #SLT0,BSEL4
MOV #125,BSEL6

```

```

;PUT ADDRESS OF SELECT REGISTER 0 IN 'ADDRESS' REG
;PUT THE DATA TO BE WRITTEN IN 'DATA' REGISTER

```

```

CMP BSEL0,#400
BEQ 1$

```

```

;ONLY 'MREQ' SHOULD BE SET
;IF IT IS, PROCEED WITH TESTING
;ELSE, SETUP FOR (& REPORT) THE ERROR

```

```

MOV BSEL0,BDATA
MOV #400,GDATA
CLR REGNUM
BR 4$

```

```

; BAD DATA
; GOOD DATA
; REG. NUMBER

```

```

CMP BSEL2,#200
BEQ 2$
MOV BSEL2,BDATA
MOV #200,GDATA
MOV #2,REGNUM
BR 4$

```

```

;'MRDY' SET? (ALSO CHECKED BY 'MSTCLR')
;YES, PROCEED WITH TESTING
; BAD DATA
; GOOD DATA
; THE REG. THAT FAILED
;EXIT TEST

```

```

CMP BSEL4,#SLT0
BEQ 3$

```

```

;COMPARE SELECT REGISTER 4 WITH THE ADDRESS SENT
;A MISMATCH IS A DEVICE FATAL ERROR

```

CVDMAA.P11 12-DEC-80 15:59

TEST 7 -- CSR, MAINTENANCE MICROCODE INTERACTION

```

4294 022240 017737 160116 002312      MOV    @SEL4,BDATA      ;GET THE BAD DATA
4295 022246 012737 000020 002310      MOV    #SLT0,GDATA     ;GET THE GOOD DATA
4296 022254 012737 000004 002334      MOV    #4,REGNUM       ;GET THE REGISTER NUMBER WHICH FAILED
4297 022262 000415                                     BR     4$
4298
4299 022264 027727 160076 000125 3$:    CMP    @SEL6,#000125    ;COMPARE SELECT REGISTER 6 WITH THE DATA SENT
4300 022272 001415                                     BEQ    60$              ;THIS PART OF THE TEST PASSES IF A MATCH IS FOUND
4301 022274 017737 160066 002312      MOV    @SEL6,BDATA     ;GET THE BAD DATA
4302 022302 012737 000125 002310      MOV    #000125,GDATA   ;GET THE GOOD DATA
4303 022310 012737 000006 002334      MOV    #6,REGNUM       ;GET THE REGISTER NUMBER
4304
4305
4306                                     ;--PREPARE FOR THE FAILURE PRINTOUT--
4307
4308 022316                                     4$:    GEDF    EM7,ERR5     ;ELSE, AN ERROR HAS BEEN FOUND
4309                                     ;          'DEVICE FATAL' ERROR # 16
4310 022316 104455                                     TRAP   C$ERDF
4311 022320 000020                                     .WORD 16
4312 022322 014564                                     .WORD EM7
4313 022324 005552                                     .WORD ERR5
4314 022326
4315 022326                                     60$:   ENDSUB
4316 022326 104403                                     L10034: TRAP   C$ESUB
4317                                     ;***** > P A R T 2 < *****
4318 022330                                     BGNSUB
4319 022330
4320 022330 104402                                     T7.2:  TRAP   C$BSUB
4321
4322 022332 112777 000002 160016      MOVB   #WRILOC,@SEL2   ;SEND THE WRITE LOCATION COMMAND
4323
4324 022340 032777 000200 160010      BIT    #200,@SEL2     ;WE SHOULD HAVE IMEDIATLY LOST 'MRDY'.
4325 022346 001421                                     BEQ    5$              ;GOT WHAT WE EXPECTED, WAIT FOR READY AGAIN
4326 022350 017737 160002 002312      MOV    @SEL2,BDATA     ;SOMETHING WRONG, SETUP FOR AND REPORT ERROR
4327 022356 004737 004434                                     JSR    PC,GETBSR       ;GET THE BSEL REGISTERS FOR DUMPING
4328 022362 012737 000002 002310      MOV    #002,GDATA     ;EXPECTED DATA
4329 022370 012737 000002 002334      MOV    #2,REGNUM      ;WE WERE TESTING BSEL2
4330 022376
4331                                     ;          'DEVICE FATAL' ERROR # 17
4332 022376 104455                                     TRAP   C$ERDF
4333 022400 000021                                     .WORD 17
4334 022402 014733                                     .WORD EM15
4335 022404 005304                                     .WORD ERR2
4336 022406
4337 022406 104410                                     ESCAPE TST
4338 022410 000206                                     TRAP   C$ESCAPE
4339                                     .WORD L10033-.
4340 022412 132777 000200 157736 5$:    BITB   #200,@SEL2     ;WAIT FOR 'MRDY' TO GO HIGH AGAIN
4341 022420 001774                                     BEQ    5$
4342
4343 022422 004737 004576                                     JSR    PC,GETWSR       ;WHEN IT DOES, GET CURRENT REGISTER CONTENTS
4344
4345 022426 023727 002246 000525      CMP    WSR0,#000525    ;COMPARE BYTE SELECT REGISTERS 0 AND 1
4346                                     ;REG 0 = 125, REG 1 = 001
4347 022434 001412                                     BEQ    6$              ;THIS PART OF THE TEST PASSES IF A MATCH IS FOUND
4348 022436 012737 000525 002310      MOV    #000525,GDATA   ;GET THE GOOD DATA
4349 022444 013737 002246 002312      MOV    WSR0,BDATA     ;GET THE BAD DATA

```



CVDMAA.P11

12-DEC-80 15:59

## TEST 7 -- CSR, MAINTENANCE MICROCODE INTERACTION

```

4350 022452 012737 000000 002334      MOV    #0,REGNUM      ;GET THE REGISTER NUMBER
4351 022460 000451                    BR     9$             ;EXIT TEST
4352
4353 022462 023727 002250 000200 6$:  CMP    WSR2,#000200   ;COMPARE BYTE SELECT REGISTERS 2 AND 3
4354                                ;REG 2 = 200 -- 'MRDY' IS SET & COMMAND IS CLEARED
4355                                ;REG 3 = 000 SHOULD BE ZEROES.
4356 022470 001412                    BEQ    7$             ;THIS PART OF THE TEST PASSES IF A MATCH IS FOUND
4357 022472 012737 000200 002310      MOV    #000200,GDATA  ;GET THE GOOD DATA
4358 022500 013737 002250 002312      MOV    WSR2,BDATA     ;GET THE BAD DATA
4359 022506 012737 000002 002334      MOV    #2,REGNUM      ;GET THE REGISTER NUMBER
4360 022514 000433                    BR     9$             ;EXIT TEST
4361
4362                                ;SUBROUTINE ATTEMPTED TO ZERO THIS LOCATION.
4363 022516 023727 002252 000020 7$:  CMP    WSR4,#SLT0     ;REG 4 = 020, THE 6502 ADDRESS TO PUT DATA
4364                                ;REG 5 = 000, ZEROED BY MSTCLR
4365 022524 001412                    BEQ    8$             ;THIS PART OF THE TEST PASSES IF A MATCH IS FOUND
4366 022526 012737 000020 002310      MOV    #SLT0,GDATA    ;GET THE GOOD DATA
4367 022534 013737 002252 002312      MOV    WSR4,BDATA     ;GET THE BAD DATA
4368 022542 012737 000004 002334      MOV    #4,REGNUM      ;GET THE REGISTER NUMBER
4369 022550 000415                    BR     9$             ;EXIT TEST
4370
4371 022552 023727 002254 000125 8$:  CMP    WSR6,#000125   ;REG 6 = 125, THE WRITE DATA
4372 022560 001415                    BEQ    ENDT7          ;THIS PART OF THE TEST PASSES IF A MATCH IS FOUND
4373 022562 012737 000125 002310      MOV    #000125,GDATA  ;GET THE GOOD DATA
4374 022570 013737 002254 002312      MOV    WSR6,BDATA     ;GET THE BAD DATA
4375 022576 012737 000006 002334      MOV    #6,REGNUM      ;GET THE REGISTER NUMBER
4376                                ;REG 7 = 000, ZEROED BY MSTCLR.
4377
4378                                ;--PREPARE FOR THE FAILURE PRINTOUT--
4379
4380 022604 9$:  GEDF    EM7,ERR5     ;REPORT ERROR.
4381                                ;      "DEVICE FATAL" ERROR # 18
4382 022604 104455                    TRAP   C$ERDF
4383 022606 000022                    .WORD 18
4384 022610 014564                    .WORD EM7
4385 022612 005552                    .WORD ERR5
4386 022614                    ENDT7:  ENDSUB
4387 022614 104403                    L10035: TRAP   C$ESUB
4388 022616                    ENDTST
4389 022616
4390 022616
4391 022616 104401                    L10033: TRAP   C$ETST

```





CVDMAA.P11 12-DEC-80 15:59

TEST 8 -- RUN FLIP-FLOP

```

4448
4449
4450
4451
4452
4453
4454
4455
4456 022736 013701 002320
4457 022742 132777 000200 157406 12$: MOV DELAY2,R1 ;SETUP AND WAIT FOR A WHILE.
4458 022750 001042 BITB #MRDY,@BSEL2 ;WHILE WE'RE WAITING, WE MAY AS WELL CHECK 'MRDY'
4459 BNE 14$ ;IF IT GETS SET, WE HAVE AN ERROR BECAUSE
4460 ;NOTHING WAS SUPPOSED TO HAPPEN WITHIN
4461 022752 105777 157374 TSTB @BSELO ;THE 6502 MICRO-PROCESSOR
4462 022756 001063 BNE 15$ ;WHILE WE'RE AT IT, WE MAY AS WELL LOOK AT
4463 022760 077110 SOB R1,12$ ;BSELO. THAT ALSO ISN'T SUPPOSED TO CHANGE.
4464 ;DECREMENT AND CHECK COUNTER -- LOOP TILL DONE
4465
4466 ;IF EVERYTHING GOES OK, WE SHOULD FALL OUT OF THE LOOP TO HERE. OTHERWISE,
4467 ;'MRDY' OR BSELO COULD CHANGE SENDING US TO '14$' OR '15$' RESPECTIVELY TO
4468 ;PRINT AN APPROPRIATE (WE HOPE) ERROR MESSAGE.
4469
4470 ;IF WE DO GET TO HERE, WE CAN NOW SET 'RUN' AND THE MLOOP SHOULD PERFORM THE
4471 ;REQUESTED FUNCTION.
4472 022762 152777 000200 157364 BISB #RUN,@BSEL1 ;SET 'RUN' AND ALLOW THE 6502 TO RUN AGAIN
4473
4474 ;NOW ALL WE HAVE TO DO IS WAIT AGAIN AS BEFORE. EXCEPT THAT THIS TIME 'MRDY'
4475 ;OR BSELO GETTING SET IS THE VALID CONDITION -- NOT THE ERROR. FAILURE TO
4476 ;PERFORM IS NOW THE ERROR WE'RE LOOKING FOR.
4477
4478 022770 013701 002320
4479 022774 132777 000200 157354 13$: MOV DELAY2,R1 ;SETUP AND WAIT FOR A WHILE.
4480 023002 001070 BITB #MRDY,@SEL2 ;WHILE WE'RE WAITING, 'MRDY' SHOULD GO NON-ZERO
4481 BNE 24$ ;IF IT GETS SET, WE CAN ASSUME THAT SOMETHING
4482 ;COMPLETED. AT LEAST WE WERE ABLE TO GET THE
4483 023004 077105 SOB R1,13$ ;6502 MICRO-PROCESSOR RUNNING AGAIN
4484 ;DECREMENT AND CHECK COUNTER -- LOOP TILL DONE
4485
4486 ;IF WE GET HERE, WE WEREN'T ABLE TO RESTORE THE 6502 TO A RUNNING STATE!
4487 023006 117737 157344 002312 MOVB @SEL2,BDATA ;SETUP FOR THE ERROR MESSAGE -- GET BAD DATA
4488 023014 004737 004434 JSR PC,GETBSR ;GET THE BSEL REGISTERS FOR DUMPING
4489 023020 113737 002312 002310 MOVB BDATA,GDATA ;PICK THE REGISTER'S DATA. THE ONLY DIFFERENCE
4490 023026 152737 000200 002310 BISB #MRDY,GDATA ;BETWEEN GOOD & BAD IS THE 'MRDY' BIT
4491 023034 012737 000002 002334 MOV #2,REGNUM ;INDICATE THAT WE'RE LOOKING AT BSEL2
4492 023042 GEDF EM17,ERR2 ;NOW REPORT THE ERROR
4493 ; 'DEVICE FATAL' ERROR # 20
4494 023042 104455 TRAP C$ERDF
4495 023044 000024 .WORD 20
4496 023046 015110 .WORD EM17
4497 023050 005304 .WORD ERR2
4498 023052 ESCAPE TST ;EXIT TEST (OR LOOP, MAYBE?)
4499 023052 104410 TRAP C$ESCAPE
4500 023054 000110 .WORD L10036-.
4501
4502
4503

```

;IF WE GET HERE, BSEL2 CHANGED WHEN THE 6502 WASN'T SUPPOSED TO BE RUNNING!

CVDMAA.P11 12-DEC-80 15:59

TEST 8 -- RUN FLIP-FLOP

```

4504 023056 117737 157274 002312 14$:  MOVB  @BSEL2,BDATA ;GET THE UNEXPECTEDLY ALTERED CONTENTS OF BSEL2
4505 023064 004737 004434          JSR   PC,GETBSR ;GET THE BSEL REGISTERS FOR DUMPING
4506 023070 113737 002312 002310  MOVB  BDATA,GDATA ;PICK THE REGISTER'S DATA. THE ONLY DIFFERENCE
4507 023076 142737 000200 002310  BICB  #MRDY,GDATA ;BETWEEN GOOD & BAD IS THE 'MRDY' BIT
4508 023104 012737 000002 002334  MOV   #2,REGNUM ;INDICATE THAT WE'RE LOOKING AT BSEL2
4509 023112          GEDF  EM17A,ERR2 ;NOW REPORT THE ERROR
4510          ; 'DEVICE FATAL' ERROR # 21
4511 023112 104455          TRAP  C$ERDF
4512 023114 000025          .WORD 21
4513 023116 015170          .WORD EM17A
4514 023120 005304          .WORD ERR2
4515 023122          ESCAPE TST ;EXIT TEST (OR LOOP, MAYBE?)
4516 023122 104410          TRAP  C$ESCAPE
4517 023124 000040          .WORD L10036-.
4518
4519          ;IF WE GET HERE, BSELO CHANGED WHEN THE 6502 WASN'T SUPPOSED TO BE RUNNING!
4520
4521 023126 117737 157220 002312 15$:  MOVB  @BSELO,BDATA ;GET THE UNEXPECTEDLY ALTERED CONTENTS OF BSELO
4522 023134 004737 004434          JSR   PC,GETBSR ;GET THE BSEL REGISTERS FOR DUMPING
4523 023140 105037 002310          CLRB  GDATA ;IT WAS SUPPOSED TO STAY AT ZERO
4524 023144 105037 002334          CLRB  REGNUM ;INDICATE THAT WE'RE LOOKING AT BSELO
4525 023150          GEDF  EM17A,ERR2 ;NOW REPORT THE ERROR
4526          ; 'DEVICE FATAL' ERROR # 22
4527 023150 104455          TRAP  C$ERDF
4528 023152 000026          .WORD 22
4529 023154 015170          .WORD EM17A
4530 023156 005304          .WORD ERR2
4531 023160          ESCAPE TST ;EXIT TEST (OR LOOP, MAYBE?)
4532 023160 104410          TRAP  C$ESCAPE
4533 023162 000002          .WORD L10036-.
4534
4535          ;IF WE GET HERE, THE TEST APPEARS TO HAVE PASSED WITH FLYING COLOURS
4536
4537 023164          24$:  ENDTST
4538 023164
4539 023164 104401          L10036: TRAP  C$ETST

```



CVDMAA.P11 12-DEC-80 15:59

TEST 9 -- LOW RAM (00-0F) SCRATCHPAD

.SBTTL TEST 9 -- LOW RAM (00-0F) SCRATCHPAD

4540  
4541  
4542  
4543  
4544  
4545  
4546  
4547  
4548  
4549  
4550  
4551  
4552  
4553  
4554  
4555  
4556  
4557  
4558  
4559  
4560  
4561  
4562  
4563  
4564  
4565  
4566  
4567  
4568  
4569  
4570  
4571  
4572  
4573  
4574  
4575  
4576  
4577  
4578  
4579  
4580  
4581  
4582  
4583  
4584  
4585  
4586  
4587  
4588  
4589  
4590  
4591  
4592  
4593  
4594  
4595

023166  
023166 004737 003762  
023172 103003  
023174 104460  
023176 104410  
023200 000152  
023202 012737 000001 002444  
023210 012737 003777 002474  
023216 004737 023354  
023222 005037 002450  
023226 005037 002452  
  
023232 004737 023372  
023236 103003  
023240 104460  
023242 104410  
023244 000106  
023246 005237 002464  
023252 023737 002464 002474  
023260 101764  
023262 104422  
  
023264 004737 023354  
023270 004737 023570  
023274 103001  
023276

```

+*****
*
*   TEST 9 -- LOW RAM (00-0F) SCRATCHPAD
*
* THIS TEST FIRST PERFORMS AN ADDRESSING TEST OF RAM LOCATIONS (00-0F), BY
* WRITING THE ADRS INTO EACH LOCATION AND AFTER EACH WRITE, ALL THE LOCATIONS
* ARE READ AND CHECKED FOR EXPECTED CONTENTS.
*
* THEN, THE TEST PERFORMS READ/WRITE DATA TESTING OF RAM LOCATIONS 00-0F,
* BY WRITING, READING, AND COMPARING ALL BYTES OF DATA PATTERN B IN EACH
* LOCATION.
*   DATA PATTERN B = 125, 252, 000, 377, 001, 002, 004, 010, 020, 040, 100,
*                   200, 376, 375, 373, 367, 357, 337, 277, 177, 000
*
+*****

```

```

:
:   BGNTST
:
:   T9::
:   JSR   PC,MSTCLR   ;INIT DMV & ENTER M-LOOP
:   BCC   1$          ;IF NO ERROR, PROCEED WITH TESTING
:   ERROR ;ELSE, REPORT ERROR
:
:   ESCAPE TST       ; & EXIT TEST
:
:   TRAP  C$ERROR
:   .WORD C$ESCAPE
:   L10037-.
:
:   1$:   MOV   #1,TMP2   ;DATA GENERATION ALGORITHM CODE
:         MOV   #2047.,TMPE ;LAST VALID ADDRESS
:   2$:   JSR   PC,T9.RST ;RESET TMP3, TMPA, & TMPF
:         CLR   TMP4      ;TEST DATA
:         CLR   TMP5      ;ACTUAL DATA
:
: ; IN THIS PHASE OF TESTING WE WRITE, READ & CHECK EACH LOCATION INDIVIDUALLY.
:
:   4$:   JSR   PC,WRCRAM ;WRITE, READ, & CHECK 1 BYTE OF RAM
:         BCC   5$          ;IF NO ERROR, PROCEED
:         ERROR ;ELSE, REPORT IT
:
:   ESCAPE TST       ; & LOOP IF ERROR
:
:   TRAP  C$ERROR
:   .WORD C$ESCAPE
:   L10037-.
:
:   5$:   INC   TMPA      ;POINT TO NEXT LOCATION
:         CMP   TMPA,TMPE ;HAVE WE TESTED ALL OF RAM?
:         BLOS 4$         ;NO, TEST ANOTHER BYTE
:         BREAK ;ELSE, SEE IF A 'C HAS BEEN STRUCK
:
:   TRAP  C$SRK
:   ;THEN PROCEED TO THE NEXT PHASE OF TESTING
:
: ; IN THIS PHASE OF TESTING WE READ & CHECK DATA WHICH SHOULD ALREADY BE IN
: ; EACH LOCATION OF RAM BEING CHECKED.
:
:   8$:   JSR   PC,T9.RST ;RESET TMP3, TMPA, & TMPF
:         JSR   PC,RGRAM ;READ & CHECK 1 BYTE OF RAM
:         BCC   9$          ;IF NO ERROR, PROCEED
:         ERROR ;ELSE, REPORT IT

```

CVDMAA.P11 12-DEC-80 15:59

TEST 9 -- LOW RAM (00-0F) SCRATCHPAD

```

4596 023276 104460                                TRAP  C$ERROR
4597 023300 005237 002464 9$: INC  TMPA      ;POINT TO NEXT LOCATION
4598 023304 023737 002464 002474  CMP  TMPA, TMPE ;HAVE WE TESTED ALL OF RAM?
4599 023312 101766      BLOS  8$      ;NO, TEST ANOTHER BYTE
4600 023314      BREAK ;ELSE, SEE IF A ^C HAS BEEN STRUCK
4601 023314 104422                                TRAP  C$BRK
4602      ;THEN PROCEED TO THE NEXT PHASE OF TESTING
4603
4604 023316 005037 007104      CLR  ER47CT   ;RESET ERROR PRINT COUNT
4605 023322 005237 002444      INC  TMP2     ; ADVANCE TO NEXT DATA GEN. ALGORITHM CODE
4606 023326 023727 002444 000007  CMP  TMP2, #7 ;HAVE WE DONE ALL THE CODES WE'RE GOING TO DO?
4607 023334 002730      BLT  2$      ;NO, THEN GO DO THIS PATTERN IN RAM
4608 023336 004537 004322      JSR  R5,WRITEI ;ELSE, CLEAR RAM LOCATION 00B3 (HEX) & EXIT
4609 023342 000173      173      ; (THIS CONVERTS TO 00B3 HEX.)
4610 023344 000000      0        ; (THIS WE HOPE, WILL CLEAR IT)
4611 023346 103001      BCC  .+4     ;IF NO ERROR, PROCEED
4612 023350      ERROR ;ELSE, REPORT IT
4613 023350 104460                                TRAP  C$ERROR
4614 023352
4615 023352
4616 023352 104401                                L10037: TRAP  C$SETST
4617
4618      ; RESET THE FOLLOWING THREE REGISTERS
4619
4620 023354 005037 002446  T9.RST: CLR  TMP3      ;TEST DATA PATTERN INDEX
4621 023360 005037 002464      CLR  TMPA      ;RAM LOCATION ADDRESS
4622 023364 005037 002476      CLR  TMPF      ;RESET ALL ERROR FLAGS
4623 023370 000207      RTS   PC
4624
4625      ; WRITE, READ, & CHECK ONE LOCATION
4626
4627 023372 010046  WRCRAM: MOV  R0, -(SP) ;SAVE WORKING REGISTERS
4628
4629 023374 004737 023750      JSR  PC, PATGEN ;GENERATE ONE DATA PATTERN BYTE
4630
4631 023400 013700 002464      MOV  TMPA, R0 ;GET ADDRESS WHERE WE CAN CHECK IT MORE EASILY
4632 023404 020027 000020      CMP  R0, #SLT0 ;IS ADDRESS BELOW THE SELECT REGISTER AREA?
4633 023410 103412      BLO  2$      ;YES, GOOD. IT CAN BE TESTED.
4634 023412 020027 000030      CMP  R0, #SLT0+8. ;IS IT ABOVE THE SELECT REGISTER AREA?
4635 023416 103007      BHIS  2$      ;YES, GOOD. IT CAN BE TESTED.
4636 023420 023727 002444 000006  CMP  TMP2, #6 ;NO, IF 'INCREMENTAL', BACK UP INDEX
4637 023426 001055      BNE  12$     ;ELSE JUST BYPASS TEST
4638 023430 005337 002446      DEC  TMP3    ;DECREMENT INDEX TO WHAT IT WAS BEFORE 'PATGEN'
4639 023434 000452      BR   12$    ; AND THEN BYPASS THE TESTING
4640
4641 023436 010037 023452  2$: MOV  R0, 4$ ;SETUP ALL POINTERS FOR THE CURRENT RAM LOCATION
4642 023442 010037 023464      MOV  R0, 8$
4643
4644 023446 004537 004310      JSR  R5, WRITE ;WRITE ONE BYTE OF THE TEST DATA
4645 023452 000000  4$: .WORD 0 ;**** MODIFIED FROM ABOVE ****
4646 023454 002450      TMP4 ;TEST DATA IS IN TMP4
4647 023456 103442      BCS  14$     ;IF ERROR WRITING, FORGET THE REST
4648
4649 023460 004537 004064      JSR  R5, READ ;READ THAT BYTE BACK AGAIN
4650 023464 000000  8$: .WORD 0 ;**** MODIFIED FROM ABOVE ****
4651 023466 002452      TMP5

```



CVDMAA.P11 12-DEC-80 15:59

TEST 9 -- LOW RAM (00-0F) SCRATCHPAD

```

4652 023470 103435          BCS      14$      ;IF ERROR READING, FORGET THE REST
4653
4654 023472 123737 002450 002452  CMPB    TMP4,TMP5  ;DID WE READ WHAT WE WROTE?
4655 023500 001430          BEQ     12$      ;YES, EXIT
4656 023502 132737 000002 002476  BITB    #BIT1,TMPF ;NO, HAVE WE ALREADY DONE THIS ERROR'S HEADER?
4657 023510 001020          BNE     9$      ;YES, ONLY REPORT DATA
4658 023512 112737 000002 002476  MOVB    #BIT1,TMPF ;ELSE, CALL MONITOR & PRINT HEADING
4659 023520          GTDF    EM47A,ERR47 ;QUEUE UP THE ERROR MESSAGE
4660
4661 023520 012737 000001 002236          ;      QUEUE "DEVICE FATAL" ERROR # 23
4662 023526 012737 000027 002240          MOV     #T.EDF,ERRTYP
4663 023534 012737 015675 002242          MOV     #23,ERRNBR
4664 023542 012737 006724 002244          MOV     #EM47A,ERRMSG
4665 023550 000402          MOV     #ERR47,ERRBLK
4666
4667 023552 004737 007110 9$:      JSR     PC,ERR47. ;JUST PRINT DATA
4668 023556 000261 10$:      SEC          ;      & SET THE ERROR FLAG
4669 023560 000401          BR      14$      ;      & GO DIRECTLY TO THE EXIT 'RTS'
4670
4671 023562 000241 12$:      CLC          ;NORMAL EXIT - MAKE SURE THE ERROR FLAG IS CLEAR
4672 023564 012600 14$:      MOV     (SP)+,R0 ;RESTORE WORK REGISTERS
4673 023566 000207          RTS     PC
4674
4675 023570 010046  RCRAM:  MOV     R0,-(SP) ;SAVE WORKING REGISTERS
4676
4677 023572 004737 023750          JSR     PC,PATGEN ;GENERATE ONE DATA PATTERN BYTE
4678
4679 023576 013700 002464          MOV     TMPA,R0   ;GET ADDRESS WHERE WE CAN CHECK IT MORE EASILY
4680 023602 020027 000020          CMP     R0,#SLT0 ;IS ADDRESS BELOW THE SELECT REGISTER AREA?
4681 023606 103412          BLO     2$      ;YES, GOOD. IT CAN BE TESTED.
4682 023610 020027 000030          CMP     R0,#SLT0+8. ;IS IT ABOVE THE SELECT REGISTER AREA?
4683 023614 103007          BHIS    2$      ;YES, GOOD. IT CAN BE TESTED.
4684 023616 023727 002444 000006          CMP     TMP2,#6   ;NO, IF "INCREMENTAL", BACK UP INDEX
4685 023624 001046          BNE     12$      ;ELSE JUST BYPASS TEST
4686 023626 005337 002446          DEC     TMP3      ;DECREMENT INDEX TO WHAT IT WAS BEFORE 'PATGEN'
4687 023632 000443          BR      12$      ; AND THEN BYPASS THE TESTING
4688
4689 023634 010037 023644 2$:      MOV     R0,8$    ;SETUP POINTER FOR THE CURRENT RAM LOCATION
4690
4691 023640 004537 004064          JSR     R5,READ   ;READ THAT BYTE BACK AGAIN
4692 023644 000000 8$:      .WORD  0        ;**** MODIFIED FROM ABOVE ****
4693 023646 002452          TMP5
4694 023650 103435          BCS     14$      ;IF ERROR READING, FORGET THE REST
4695
4696 023652 123737 002450 002452  CMPB    TMP4,TMP5  ;WAS THIS LOC. STILL OK?
4697 023660 001430          BEQ     12$      ;YES, EXIT
4698 023662 132737 000004 002476  BITB    #BIT2,TMPF ;NO, HAVE WE ALREADY DONE THIS ERROR'S HEADER?
4699 023670 001020          BNE     9$      ;YES, ONLY REPORT DATA
4700 023672 112737 000004 002476  MOVB    #BIT2,TMPF ;ELSE, CALL MONITOR & PRINT HEADING
4701 023700          GTDF    EM47B,ERR47 ;QUEUE UP THE ERROR MESSAGE
4702
4703 023700 012737 000001 002236          ;      QUEUE "DEVICE FATAL" ERROR # 24
4704 023706 012737 000030 002240          MOV     #T.EDF,ERRTYP
4705 023714 012737 015735 002242          MOV     #24,ERRNBR
4706 023722 012737 006724 002244          MOV     #EM47B,ERRMSG
4707 023730 000402          MOV     #ERR47,ERRBLK
4708
4709          BR      10$

```

CVDMAA.P11 12-DEC-80 15:59

TEST 9 -- LOW RAM (00-0F) SCRATCHPAD

```

4708
4709 023732 004737 007110
4710 023736 000261
4711 023740 000401
4712
4713 023742 000241
4714 023744 012600
4715 023746 000207
4716
4717
4718
4719
4720
4721
4722
4723
4724
4725
4726
4727
4728
4729
4730
4731
4732
4733
4734
4735
4736
4737
4738
4739
4740
4741
4742

```

```

9$: JSR PC,ERR47. ;JUST PRINT DATA
10$: SEC ; & SET THE ERROR FLAG
BR 14$ ; & GO DIRECTLY TO THE EXIT 'RTS'

12$: CLC ;NORMAL EXIT - MAKE SURE THE ERROR FLAG IS CLEAR
14$: MOV (SP)+,R0 ;RESTORE WORK REGISTERS
RTS PC

```

```

:*****
: PATGEN -- SUBROUTINE TO GENERATE A TEST DATA BYTE FOR A SPECIFIC ELEMENT
:
: CALLING SEQUENCE:
:
: <SET TEST PATTERN CODE # IN 'TMP2'>
: <SET TEST PATTERN INDEX IN 'TMP3'>
: JSR PC,PATGEN
: <NEXT SEQUENTIAL INSTRUCTION>
:
: TEST PATTERN CODES:
:
: 1 -- ALL ONES
: 2 -- ALL ZEROES
: 3 -- 1 BIT ALTERNATING
: 4 -- 2 BITS ALTERNATING
: 5 -- ADDRESS IN ADDRESS
: 6 -- INCREMENTAL (INDEX IN ADDRESS)
:
: THE TEST PATTERN INDEX INDICATES HOW FAR INTO THE TEST PATTERN STRING OF
: BYTES WE ARE. I.E. IT SPECIFIES THE NUMBER OF THE BYTE OF THE WHOLE STRING
: OF BYTES COMPOSING THE COMPLETE TEST PATTERN.
:*****

```

```

4743 023750
4744 023750 023727 002444 000002
4745 023756 002414
4746 023760 001417
4747 023762 023727 002444 000004
4748 023770 002416
4749 023772 001431
4750 023774 023727 002446 000006
4751 024002 002441
4752 024004 001444
4753 024006 000404
4754
4755 024010 112737 000377 002450
4756 024016 000443
4757
4758 024020 105037 002450
4759 024024 000440
4760
4761 024026 132737 000001 002446
4762 024034 001404
4763 024036 112737 000125 002450

```

```

PATGEN:
CMP TMP2,#2 ;DECODE THE TEST PATTERN IDENTIFIER
BLT 1$ ;0, 1, OR NEGATIVE WILL GIVE 'ALL ONES'
BEQ 2$ ;2 = 'ALL ZEROES'
CMP TMP2,#4 ;3 = '1 BIT ALTERNATING'
BLT 3$ ;4 = '2 BIT ALTERNATING PATTERN'
BEQ 4$
CMP TMP3,#6 ;5 = 'ADDRESS IN ADDRESS'
BLT 5$ ;6 = 'INCREMENTAL' (INDEX IN ADDRESS)
BEQ 6$ ;UNDEFINED = 'ALL ZEROES'
BR 2$

1$: MOVB #377,TMP4 ;'ALL ONES' DATA PATTERN
BR 60$

2$: CLRB TMP4 ;'ALL ZEROES' DATA PATTERN
BR 60$

3$: BITB #1,TMP3 ;'1 BIT ALTERNATING' PATTERN
BEQ 20$ ;IF EVEN, USE '252'
MOVB #125,TMP4 ;IF ODD, USE '125'

```



CVDMAA.P11 12-DEC-80 15:59

TEST 9 -- LOW RAM (00-0F) SCRATCHPAD

```

4764 024044 000430          BR      60$          ; PATTERN: 10101010
4765 024046 112737 000252 002450 20$:  MOVB   #252,TMP4
4766 024054 000424          BR      60$          ; PATTERN: 01010101
4767
4768 024056 132737 000001 002446 4$:  BITB   #1,TMP3      ; '2 BIT ALTERNATING' PATTERN
4769 024064 001404          BEQ    22$          ; IF EVEN, USE '214'
4770 024066 112737 000063 002450      MOVB   #063,TMP4    ; IF ODD, USE '063'
4771 024074 000414          BR      60$          ; PATTERN: 11001100
4772 024076 112737 000214 002450 22$:  MOVB   #214,TMP4
4773 024104 000410          BR      60$          ; PATTERN: 00110011
4774
4775 024106 113737 002464 002450 5$:  MOVB   TMPA,TMP4    ; 'ADDRESS IN ADDRESS'
4776 024114 000404          BR      60$
4777
4778 024116 113737 002446 002450 6$:  MOVB   TMP3,TMP4    ; 'INCREMENTAL' (INDEX IN ADDRESS)
4779 024124 000400          BR      60$
4780
4781 024126 005237 002446      60$:  INC    TMP3          ; INCREMENT PATTERN INDEX FOR NEXT CALL
4782 024132 000207      62$:  RTS    PC
4783
4784

```

CVDMAA.P11 12-DEC-80 15:59

TEST 10 -- DATA RAM MOVING INVERSIONS (LOC'S 0018-01FF HEX)

.SBTTL TEST 10 -- DATA RAM MOVING INVERSIONS (LOC'S 0018-01FF HEX)

4785  
4786  
4787  
4788  
4789  
4790  
4791  
4792  
4793  
4794  
4795  
4796  
4797  
4798  
4799  
4800  
4801  
4802  
4803  
4804  
4805  
4806  
4807  
4808  
4809  
4810  
4811  
4812  
4813  
4814  
4815  
4816  
4817  
4818  
4819  
4820  
4821  
4822  
4823  
4824  
4825  
4826  
4827  
4828  
4829  
4830  
4831  
4832  
4833  
4834  
4835  
4836  
4837  
4838  
4839  
4840

```

*****
*
* TEST 10 -- DATA RAM MOVING INVERSIONS (LOC'S 0018-01FF HEX)
*
* GENERAL DESCRIPTION:
* FIRST, THE 2K BYTE LOCATIONS IN RAM ARE LOADED WITH 0'S (SEE NOTE BELOW).
* THEN, THE FIRST LOCATION IS READ AND CHECKED, A SINGLE 1 IS WRITTEN INTO
* THE LOW BIT POSITION, AND THIS IS READ AND CHECKED. THIS IS DONE FOR ALL
* BYTES IN THE RAM, BY INCREMENTING THE ADDRESS TO POINT TO THE NEXT RAM
* LOCATION.
* THEN, THE NEXT BIT POSITION IS CHOSEN TO INSERT A 1, AND ALL LOCATIONS
* ARE READ, WRITTEN, AND READ AS BEFORE. THIS IS CONTINUED FOR ALL BIT
* POSITIONS UNTIL THE ENTIRE RAM IS WRITTEN TO ALL 1'S. THE ABOVE OPERATIONS
* ARE PERFORMED A SECOND TIME, WITH 0'S INSERTED INTO THE RAM INSTEAD OF 1'S.
* THIS RESULTS IN THE ENTIRE RAM BEING WRITTEN TO ALL 0'S.
* THIS TEST CONSTITUTES A THOROUGH TEST OF THE RAM. IT IS CAPABLE OF
* DETECTING THE FOLLOWING FAULTS : STUCK ADDRESS BITS, UNI- AND BI-DIRECT-
* IONAL COUPLING BETWEEN ADDRESS BITS, STUCK MEMORY BITS, AND UNI- AND
* BI-DIRECTIONAL COUPLING BETWEEN MEMORY BITS IN BOTH ROWS AND COLUMNS OF THE
* MEMORY MATRIX.
*
* NOTE:
* THIS TEST DOES NOT CHECK LOCATIONS 0010-001F, SO THAT THE PRIMARY CSR'S
* ARE NOT WRITTEN. IT DOES TEST LOCATIONS 0000-000F (SCRATCHPAD RAM) AND
* LOCATIONS 0020-002F (SECONDARY CSR'S), AS WELL AS 0030-0800 (BASIC RAM).
*
* THE 'TMPA' REGISTERS ARE USED HERE TO CONTAIN THE VARIOUS CONSTANTS &
* VARIABLES USED THROUGHOUT THIS TEST. A LIST OF THEIR ASSIGNMENTS SEEMS
* USEFUL SO HERE IT IS:
*
* TMP0 POINTS TO THE FIRST LOCATION AFTER THE SELECT REGISTERS.
*
* TMP1 ----
*
* TMP2 TEST PATTERN ID CODE -- UNUSED BY THIS TEST.
*
* TMP3 TEST DATA PATTERN INDEX -- UNUSED BY THIS TEST.
*
* TMP4 TEST DATA PATTERN. THE HIGH BYTE IS THE PATTERN BEING WRITTEN
* ON ANY GIVEN PASS AND THE LOW BYTE IS THE PATTERN THAT WAS
* WRITTEN BY THE PREVIOUS PASS THROUGH THE RAM.
*
* TMP5 DATA READ FROM THE RAM. ONLY THE LOW BYTE IS USED.
*
* TMP6 ----
*
* TMP7 ----
*
* TMP8 ----
*
* TMP9 ----
*
* TMPA RAM ADDRESS BEING TESTED.
*
* TMPB BIT POINTER. NUMBER OF THE BIT WITHIN THE DATA FIELD WHICH IS
* BEING SWITCHED ON EACH WRITE WITHIN THE CURRENT PASS.
*
*****

```



CVDMAA.P11 12-DEC-80 15:59

TEST 10 -- DATA RAM MOVING INVERSIONS (LOC'S 0018-01FF HEX)

4841  
4842  
4843  
4844  
4845  
4846  
4847  
4848  
4849  
4850  
4851  
4852  
4853  
4854  
4855  
4856  
4857  
4858  
4859  
4860  
4861  
4862  
4863  
4864  
4865  
4866  
4867  
4868  
4869  
4870  
4871  
4872  
4873  
4874  
4875  
4876  
4877  
4878  
4879  
4880  
4881  
4882  
4883  
4884  
4885  
4886  
4887  
4888  
4889  
4890  
4891  
4892  
4893  
4894  
4895  
4896

024134  
024134 004737 003762  
024140 103003  
024142  
024142 104460  
024144  
024144 104410  
024146 000744  
024150  
  
024150 012737 000030 002440  
024156 012737 003777 002474  
024164 005037 002462  
024170 005037 002476  
024174 012737 177777 002470  
  
024202 005037 024216  
024206 012703 000020  
  
024212 004537 004322  
024216 000000  
024220 000000  
024222 103003  
024224  
024224 104460  
024226  
024226 104410  
024230 000662  
024232 005237 024216

```

: *      TMPC      DATA FLAG. BIT 0 OF THIS WORD IS THE VALUE TO WHICH THE BIT
: *      IDENTIFIED IN TMPB IS BEING SET ON EACH WRITE IN THE CURRENT
: *      PASS.
: *
: *      TMPD      DIRECTION SWITCH. 0 = FORWARD    NON-ZERO = BACKWARD
: *
: *      TMPE      LAST VALID ADDRESS TO BE TESTED. (I.E. THE END OF RAM)
: *
: *      TMPF      ERROR FLAGS. BIT 1 SET = THE LAST DETECTED ERROR WAS THE READ
: *      OF THE PREVIOUS DATA BEFORE WRITING THE NEW DATA. IF BIT2 IS
: *      SET, THE READ AFTER WRITE FAILED. IF EITHER IS SET WHEN AN
: *      ERROR IS DETECTED, THE SUPERVISOR IS NOT CALL'D AND THEREFOR
: *      IT'S ERROR COUNTER WILL NOT REFLECT THE ERROR -- INSTEAD, THE
: *      DATA LINE IS PRINTED. (UNLESS THE ERROR HANDLER'S DATA LINE
: *      PRINT COUNT HAS EXCEEDED ITS LIMIT -- IN WHICH CASE ITS
: *      INVOCATION IS IGNORED.)
: *
: *****
:
:      BGNTST
:
:      JSR      PC,MSTCLR      ;INIT DMV & ENTER M-LOOP
:      BCC      1$            ;IF NO ERROR, PROCEED WITH TESTING
:      ERROR    ;ELSE, REPORT ERROR
:
:      ESCAPE  TST            ; & EXIT TEST
:
:      TRAP    C$ERROR
:      TRAP    C$ESCAPE
:      .WORD   L10040-
:
1$:
:===== ACTUAL MOVING INVERSIONS ALGORITHM =====
:----- INITIALIZE OUTER LOOP -----
:
:      MOV      #24.,TMP0      ;INIT. POINTER TO 1'ST RAM LOC. AFTER SEL REG'S
:      MOV      #2047.,TMPE    ;IDENTIFY LAST ADDRESS TO BE TESTED
:      CLR      TMP9
:      CLR      TMPF          ;ERROR FLAG -- INDICATE NO ERRORS YET
:      MOV      #-1.,TMPC     ;DATA = 1'S FIRST
:
:----- INITIALIZE THE AREA BEING TESTED BY CLEARING IT TO ZEROES -----
:
:      ZERO OUT LOCATIONS 0 THROUGH 10 (HEX) -- THOSE BELOW THE SELECT REGISTERS
:
:      CLR      3$            ;INITIALIZE ADDRESS
:      MOV      #SLT0,R3      ;RAM ADDRESS OF BSEL0 WILL DO AS BYTE COUNT
:
2$: JSR      R5,WRITEI        ;ZERO OUT LOC'S 0 --> 10 (HEX)
3$: .WORD   0                ; ADDRESS
:      0                    ; DATA
:      BCC      .+10         ;IF NO ERROR, PROCEED
:      ERROR    ;ELSE, REPORT IT
:
:      ESCAPE  TST            ; AND EXIT THIS TEST
:
:      TRAP    C$ERROR
:      TRAP    C$ESCAPE
:      .WORD   L10040-
:
:      INC     3$            ;POINT TO NEXT LOCATION
    
```

CVDMAA.P11 12-DEC-80 15:59

TEST 10 -- DATA RAM MOVING INVERSIONS (LOC'S 0018-01FF HEX)

```

4897 024236 077313          SOB      R3,2$          ;IF MORE TO BE DONE, DO IT
4898
4899
4900          : ZERO OUT THE REST OF RAM -- ALL LOC'S ABOVE THE SELECT REGISTERS
4901 024240 013737 002440 024264      MOV      TMP0,6$        ;FIRST LOCATION OF TEST AREA (18 HEX)
4902 024246 013703 002474          MOV      TMPE,R3        ;START WITH 'LAST ADDR. TO BE TESTED' AND CALC.
4903 024252 163703 002440          SUB      TMP0,R3        ;THE # OF LOCATIONS TO BE TESTED (800-18 (HEX))
4904 024256 005203          INC      R3            ; (THIS MAKES SURE WE GET EVERY SINGLE BYTE)
4905
4906 024260 004537 004322      4$:     JSR      R5,WRITEI ;ZERO OUT THE ALL OF THE TEST AREA
4907 024264 000000          6$:     .WORD    0
4908 024266 000000          0
4909 024270 103003          BCC     .+10          ;IF NO ERROR, PROCEED
4910 024272          ERROR          ;ELSE, REPORT IT
4911 024272 104460          ESCAPE  TST          ; AND EXIT THIS TEST          TRAP    C$ERROR
4912 024274          .WORD    L10040-.
4913 024274 104410          TRAP    C$ESCAPE
4914 024276 000614          .WORD    L10040-.
4915 024300 005237 024264          INC      6$          ;POINT TO NEXT LOCATION
4916 024304 077313          SOB     R3,4$        ;IF MORE TO BE DONE, DO IT
4917 024306 105037 002450          CLRB   TMP4         ;THIS IS WHAT WE JUST SET ALL RAM LOCATIONS TO
4918
4919          ;----- BEGINNING OF OUTER LOOP -----
4920
4921 024312 005037 002472      8$:     CLR      TMPD          ;'SET FWD SEQUENCE' (DIRECTION FLAG)
4922 024316 005037 002466          CLR      TMPB          ;'SET BIT POSITION = 0' (BIT POINTER)
4923          ;'SET ADDRESS = 0' BUT OUR MEMORY STARTS @
4924          ; 18 HEX. SO:
4925 024322 005037 002464          CLR      TMPA          ; INITIALIZE ADDRESS POINTER
4926 024326 112737 000001 002451      MOVB   #BIT0,TMP4+1 ;INITIALIZE CURRENT & NEXT DATA BYTES
4927
4928          ;----- 'READ CURRENT ADDRESS' -----
4929
4930 024334 000240          10$:    NOP
4931 024336 000240          NOP
4932 024340          BREAK          ;FIRST SEE IF A ^C HAS BEEN STRUCK BY OPERATOR          TRAP    C$BRK
4933 024340 104422          MOV     TMPA,40$     ;NO, PUT ADDRESS INTO READ CALL
4934 024342 013737 002464 024354      JSR     R5,READ      ;GO READ ONE LOCATION
4935 024350 004537 004064          0          ;**** MODIFIED ABOVE **** (ADDRESS)
4936 024354 000000          40$:    0          ;ADDRESS OF DATA READ
4937 024356 002452          TMP5          ;IF NO ERROR, PROCEED
4938 024360 103003          BCC     .+10          ;ELSE, REPORT IT
4939 024362          ERROR
4940 024362 104460          ESCAPE  TST          ; AND EXIT THIS TEST          TRAP    C$ERROR
4941 024364          .WORD    L10040-.
4942 024364 104410          TRAP    C$ESCAPE
4943 024366 000524          .WORD    L10040-.
4944
4945          ;----- CHECK DATA (FIRST TIME) -----
4946
4947 024370 000240          NOP
4948 024372 000240          NOP
4949 024374 123737 002452 002450      CMPB   TMP5,TMP4     ;CHECK AGAINST EXPECTED DATA
4950 024402 001421          BEQ     12$          ;IF OK, PROCEED
4951 024404 032737 000006 002476      BIT    #BIT1+BIT2,TMPF ;NO, HAS AN ERROR ALREADY BEEN REPORTED?
4952 024412 001010          BNE     42$          ;YES, JUST PRINT DATA IF WANTED

```



CVDMAA.P11 12-DEC-80 15:59

TEST 10 -- DATA RAM MOVING INVERSIONS (LOC'S 0018-01FF HEX)

```

4953 024414 012737 000002 002476      MOV    #BIT1,TMPF      ;NO, SET FLAG FOR NEXT TIME
4954 024422                        GEDF   EM48A,ERR48    ; AND PRINT COMPLETE ERROR MESSAGE
4955                                     ; 'DEVICE FATAL' ERROR # 25
4956 024422 104455                        TRAP   C$ERDF
4957 024424 000031                        .WORD 25
4958 024426 016016                        .WORD EM48A
4959 024430 007632                        .WORD ERR48
4960 024432 000405
4961 024434 012737 000002 002476 42$: BR    12$      ;PROCEED WITH TESTING
4962 024442 004737 010030      MOV    #BIT1,TMPF    ;INDICATE A 'PRE' WRITE ERROR
4963                                     JSR    PC,ERR48.    ;USE ERROR HANDLER ONLY -- NO HEADER
4964                                     ;----- WRITE NEW DATA -----
4965
4966 024446 013737 002464 024460 12$: MOV    TMPA,44$      ;GET THIS ADDRESS FOR THIS WRITE CALL
4967 024454 004537 004310      JSR    R5,WRITE    ;WRITE THE UPDATED DATA IN THIS LOCATION
4968 024460 000000      44$: .WORD 0
4969 024462 002451      TMP4+1      ;NEW DATA ELEMENT RESIDES IN TMPD+1
4970 024464 103003      BCC    .+10      ;IF NO ERROR, PROCEED
4971 024466                        ERROR        ;ELSE, REPORT IT
4972 024466 104460                        ESCAPE TST      ; AND EXIT THIS TEST      TRAP   C$ERROR
4973 024470                        ;
4974 024470 104410                        ;
4975 024472 000420                        ;
4976                                     ;----- RE-'READ CURRENT ADDRESS' -----
4977
4978
4979 024474 013737 002464 024506      MOV    TMPA,46$      ;GET ADDRESS FOR THIS READ
4980 024502 004537 004064      JSR    R5,READ     ;READ DATA JUST WRITTEN
4981 024506 000000      46$: .WORD 0
4982 024510 002452      TMP5
4983 024512 103003      BCC    .+10      ;IF NO ERROR, PROCEED
4984 024514                        ERROR        ;ELSE, REPORT IT
4985 024514 104460                        ESCAPE TST      ; AND EXIT THIS TEST      TRAP   C$ERROR
4986 024516                        ;
4987 024516 104410                        ;
4988 024520 000372                        ;
4989                                     ;----- CHECK NEW DATA VALUE -----
4990
4991
4992 024522 000240      NOP
4993 024524 000240      NOP
4994 024526 123737 002451 002452      CMPB  TMP4+1,TMP5    ;DID THE WRITE WORK CORRECTLY?
4995 024534 001421      BEQ   14$          ;YES, PROCEED WITH TESTING
4996 024536 032737 000006 002476      BIT   #BIT1+BIT2,TMPF ;NO, HAS AN ERROR ALREADY BEEN REPORTED?
4997 024544 001010      BNE   48$          ;YES, ONLY USE ERROR HANDLER -- NO HEADER PLEASE
4998 024546 012737 000004 002476      MOV    #BIT2,TMPF    ;NO, INDICATE THAT WE'RE PRINTING A HEADER HERE
4999 024554                        GEDF   EM48A,ERR48    ;REPORT RE-WRITE ERROR
5000                                     ; 'DEVICE FATAL' ERROR # 26
5001 024554 104455                        TRAP   C$ERDF
5002 024556 000032                        .WORD 26
5003 024560 016016                        .WORD EM48A
5004 024562 007632                        .WORD ERR48
5005 024564 000405      BR    14$          ;PROCEED WITH TESTING
5006
5007 024566 012737 000004 002476 48$: MOV    #BIT2,TMPF    ;INDICATE A 'POST' WRITE ERROR
5008 024574 004737 010030      JSR    PC,ERR48.    ;JUST REPORT DATA -- NO HEADER

```

CVDMAA.P11 12-DEC-80 15:59

TEST 10 -- DATA RAM MOVING INVERSIONS (LOC'S 0018-01FF HEX)

```

5009
5010 ;----- 'FORWARD SEQUENCE ?' -----
5011
5012 024600 000240 14$: NOP
5013 024602 005737 002472 TST TMPD ;CHECK DIRECTION -- 0 = FORWARD
5014 024606 001056 BNE 26$ ;REVERSE ==> PROCESS REVERSE ADDRESSING
5015 ;FORWARD
5016
5017 ;----- PROCESS FORWARD SEQUENCE -- 'LAST ADDRESS' -----
5018
5019 024610 000240 16$: NOP
5020 024612 023737 002464 002474 CMP TMPA, TMPE ;WAS THIS ADDR. THE LAST ONE?
5021 024620 001413 BEQ 18$ ;YES, THEN CHECK THE BIT POSITION
5022 024622 005237 002464 50$: INC TMPA ;NO, THEN INCREMENT THE ADDR.
5023
5024 ; HERE WE MAKE SURE THE ADDRESS IS NOT WITHIN THE SELECT REGISTER AREA. IF IT
5025 ; IS, WE WON'T USE IT -- BUT GO BACK AND DECREMENT TO THE NEXT ADDRESS AGAIN.
5026
5027 024626 022737 000020 002464 51$: CMP #SLTO, TMPA ;IS IT BELOW THE AREA WE CAN'T CHECK?
5028 024634 101237 BHI 10$ ;YES, THEN WE CAN CHECK THIS LOCATION -- DO IT
5029 024636 023737 002440 002464 CMP TMP0, TMPA ;IS IT BELOW THE BOTTOM ADDRESS?
5030 024644 101633 BLOS 10$ ;NO, TEST THIS LOCATION
5031 024646 000765 BR 50$ ;YES, PERFORM THE INCREMENT AGAIN
5032
5033 ;----- 'FWD' SEQUENCE -- 'LAST BIT POSITION?' -----
5034
5035 024650 000240 18$: NOP
5036 024652 005037 010024 CLR ER48CT ;RESET ERROR PRINT COUNT
5037 024656 023727 002466 000007 CMP TMPB, #7 ;DID WE JUST PROCESS THE LAST BIT POSITION?
5038 024664 002016 BGE 20$ ;YES, THEN WERE WE DOING 1'S OR 0'S
5039 024666 005237 002466 INC TMPB ;NO, THEN INCREMENT THE BIT COUNTER
5040 024672 005037 002464 24$: CLR TMPA ;RE-INITIALIZE ADDRESS POINTER
5041 024676 113737 002451 002450 57$: MOVB TMP4+1, TMP4 ;USE 'NEXT' DATA AS 'CURRENT' DATA
5042 024704 013700 002470 MOV TMP0, R0 ;USE ONE BIT OF THE 'DATA' SWITCH TO
5043 024710 006000 ROR R0
5044 024712 106137 002451 ROLB TMP4+1 ;BUILD A NEW 'NEXT' DATA VALUE
5045 024716 000137 024334 55$: JMP 10$ ; & TEST IT
5046
5047 ;----- 'FWD' SEQUENCE -- 'DATA = 1?' -----
5048
5049 024722 000240 20$: NOP
5050 024724 005037 002466 CLR TMPB ;POINT TO BIT 0,
5051 024730 005137 002470 COM TMP0 ;SWITCH DATA. IF 1'S, DO 0'S; IF 0'S DO 1'S
5052 024734 001756 BEQ 24$ ;IF WENT TO FORWARD, .....
5053 024736 005137 002472 COM TMPD ;SWITCH DIRECTION
5054 024742 000755 BR 57$ ;ELSE, BACKWARD.....
5055
5056 ;----- 'BKWD' SEQUENCE -- 'ADDRESS = 0?' -----
5057
5058 024744 000240 26$: NOP
5059 024746 005737 002464 TST TMPA ;HAVE WE JUST PROCESSED THE FIRST ADDRESS?
5060 024752 001413 BEQ 28$ ;YES, CHECK BIT POSITION
5061 024754 005337 002464 52$: DEC TMPA ;NO, DECREMENT THE ADDRESS
5062
5063 ; HERE WE MAKE SURE THE ADDRESS IS NOT WITHIN THE SELECT REGISTER AREA. IF IT
5064 ; IS, WE WON'T USE IT -- BUT GO BACK AND DECREMENT TO THE NEXT ADDRESS AGAIN.

```



CVDMAA.P11 12-DEC-80 15:59

TEST 10 -- DATA RAM MOVING INVERSIONS (LOC'S 0018-01FF HEX)

```

5065
5066 024760 022737 000020 002464 56$:  CMP      #SLT0,TMPA      ;IS IT BELOW THE AREA WE CAN'T CHECK?
5067 024766 101031                BHI      58$          ;YES, THEN WE CAN CHECK THIS LOCATION -- DO IT
5068 024770 023737 002440 002464    CMP      TMP0,TMPA    ;IS IT BELOW THE BOTTOM ADDRESS?
5069 024776 101425                BLOS    58$          ;NO, TEST THIS LOCATION
5070 025000 000765                BR       52$          ;YES, PERFORM THE DECREMENT AGAIN
5071
5072 ;----- 'BKWD' SEQUENCE -- 'LAST BIT POSITION' -----
5073
5074 025002 000240                28$:  NOP
5075 025004 005037 010024    CLR      ER48CT      ;RESET ERROR PRINT COUNT
5076 025010 022737 000007 002466    CMP      #7,TMPB     ;LAST BIT POSITION?
5077 025016 003417                BLE     30$          ;YES, CHECK DATA
5078 025020 005237 002466    INC     TMPB         ;NO, INCREMENT BIT POINTER,
5079 025024 113737 002451 002450 29$:  MOV     TMP4+1,TMP4   ;USE 'NEXT' DATA AS 'CURRENT' DATA
5080 025032 013700 002470    MOV     TMPC,R0      ;USE ONE BIT OF THE 'DATA' SWITCH TO
5081 025036 006000                ROR     R0
5082 025040 106137 002451    ROLB   TMP4+1       ;BUILD A NEW 'NEXT' DATA VALUE
5083 025044 013737 002474 002464    MOV     TMPE,TMPA    ;
5084 025052 000137 024334    58$:  JMP      10$        ;   POINT TO LAST ADDRESS AGAIN,
;                                     ;   & TEST IT
5085
5086 ;----- 'BKWD' SEQUENCE -- 'DATA = 1?' -----
5087
5088 025056 000240                30$:  NOP
5089 025060 005137 002470    COM     TMPC         ;SWITCH DATA TYPE
5090 025064 001003                BNE     32$          ;NOW 1'S -- CHECK ADDRESS'S 'LSB'
5091 025066 005037 002466    CLR     TMPB         ;NOW 0'S -- POINT TO BIT POSITION 0 AGAIN
5092 025072 000754                BR      29$         ;   RESET ADDRESS & TEST IT
5093
5094
5095 ;----- 'STOP' -----
5096
5097 025074 000240                32$:  NOP
5098 025076 004537 004322 38$:  JSR     R5,WRITEI   ;CLEAR RAM LOCATION 00B3 (HEX) & EXIT
5099 025102 000173                173    ;   (THIS CONVERTS TO 00B3 HEX.)
5100 025104 000000                0      ;   (THIS WE HOPE, WILL CLEAR IT)
5101 025106 103001                BCC    .+4          ;IF NO ERROR, PROCEED
5102 025110                ERROR   ;ELSE, REPORT IT
5103 025110 104460                TRAP   C$ERROR
5104 025112                ENDTST ;THATS ALL FOLKS!
5105 025112                L10040:
5106 025112 104401                TRAP   C$ETST
5107
5108 ;=====
.EVEN

```

CVDMAA.P11 12-DEC-80 15:59

TEST 11 -- VIA REGISTER ADDRESSING

.SBTTL TEST 11 -- VIA REGISTER ADDRESSING

5109  
5110  
5111  
5112  
5113  
5114  
5115  
5116  
5117  
5118  
5119  
5120  
5121  
5122  
5123  
5124  
5125  
5126  
5127  
5128  
5129  
5130  
5131  
5132  
5133  
5134  
5135  
5136  
5137  
5138  
5139  
5140  
5141  
5142  
5143  
5144  
5145  
5146  
5147  
5148  
5149  
5150  
5151  
5152  
5153  
5154  
5155  
5156  
5157  
5158  
5159  
5160  
5161  
5162  
5163  
5164

```

:*****
:*
:* TEST 11 -- VIA REGISTER ADDRESSING
:*
:* VIA == '6522 VERSATILE INTERFACE ADAPTER'
:*
:* A MASTER CLEAR IS PERFORMED, NEXT, TIMER 1 LATCHES
:* ARE CLEARED BY WRITING 000 INTO VIA REGS 6 & 7
:* THEN, 377 IS LOADED INTO DATA DIRECTION REGISTERS A, B (DDRA, DDRB) TO
:* SET THE PORT PINS FOR OUTPUT MODE.
:* THEN, A DIFFERENT BYTE OF DATA PATTERN C IS WRITTEN INTO EACH VIA
:* LOCATION, (EXCEPT THE TIMER REGS 4,5,10,11 OCT) AND AFTER EACH IS WRITTEN,
:* ALL VIA REGS (EXCEPT 4,5,10,11) ARE READ AND COMPARED TO EXPECTED
:* CONTENTS. NOTE THAT SOME VIA REGS ARE ALTERED BY THE LOADING OF OTHERS,
:* AND THE PROGRAM TAKES THIS INTO ACCOUNT, IN THE SETTING OF EXPECTED REG
:* VALUES. THE DATA PATTERN IS CHOSEN TO AVOID ACTIVATING THE VIA CHIP (SUCH
:* AS GENERATING OUTPUTS ON CA1, CA2, CB1, CB2, OR CAUSING 6502
:* INTERRUPT REQUESTS).
:* DATA PATTERN C (WITH VIA REGS AND THEIR DATA SHOWN IN OCTAL) :
:* REGISTER = 00 01 02 03 06 07 12 13 14 15 16 17
:* DATA = 100, 101, 377, 377, 106, 107, 112, 040, 042, 000, 200, 117
:* NEXT, 000 IS LOADED INTO DDRA, AND DDRB IS READ AND COMPARED TO 377. THEN,
:* THE 377 IS LOADED BACK INTO DDRA, AND DDRB IS LOADED WITH 000 AND DDRA IS
:* READ AND COMPARED TO 377.

```

```

:*****
:
: BGNTST
:
: T11::
: JSR PC,MSTCLR ;INIT DMV AND START UP THE MAINT. LOOP
: BCC 1$ ;IF NO ERROR, PROCEED
: ERROR ;ELSE, REPORT IT AND
: TRAP C$ERROR
: BR 25$ ; EXIT THIS CLEAR
:
: 1$: JSR R5,WRITEI ;CLEAR THE TIMER 1 LATCHES
: TILL
: 0
: BCC 30$ ;IF AN ERROR OCCURED,
: ERROR ;REPORT IT &
: TRAP C$ERROR
: BR 25$ ; EXIT
: 30$: JSR R5,WRITEI
: T1LH
: 0
: BCC 31$ ;IF AN ERROR OCCURED,
: ERROR ;REPORT IT &
: TRAP C$ERROR
: BR 25$ ; EXIT
:
: ; LOAD UP THE VIA'S REGISTERS WITH THE FIXED DATA STREAM OF PATTERN 'C'
: 31$: MOV PATC,R3 ;GET COUNT OF # OF WRITES TO BE PERFORMED
: MOV #PATC+2,R2 ;SETUP POINTER TO REGISTER ADDRESSES & DATA

```

```

025114
025114 004737 003762
025120 103002
025122 104460
025124 000546
025126 004537 004322
025132 120006
025134 000000
025136 103002
025140 104460
025142 000537
025144 004537 004322
025150 120007
025152 000000
025154 103002
025156 104460
025160 000530
025162 013703 002556
025166 012702 002560

```



CVDMAA.P11 12-DEC-80 15:59

TEST 11 -- VIA REGISTER ADDRESSING

```

5165
5166 025172 012737 120000 025214 2$: MOV #ORB,4$ ;ADDRESS OF FIRST REGISTER
5167 025200 152237 025214 ;OR IN REGISTER # TO BUILD REGISTER ADDRESS
5168 025204 112237 025216 MOVB (R2)+,5$ ;THIS IS THE DATA WE WANT TO WRITE
5169
5170 025210 004537 004322 JSR R5,WRITEI ;WRITE ONE REGISTER WITH THE DESIRED DATA
5171 025214 000000 4$: 0 ;*** MODIFIED FROM ABOVE *** DESTINATION ADDR.
5172 025216 000000 5$: 0 ;*** MODIFIED FROM ABOVE *** DATA
5173
5174 025220 103002 BCC 32$ ;IF AN ERROR OCCURED,
5175 025222 ERROR ;REPORT IT &
5176 025222 104460 TRAP C$ERROR
5177 025224 000506 ; EXIT
5178 025226 077317 32$: SOB R3,2$ ;LOOP UNTIL THE WHOLE TABLE HAS BEEN WRITTEN
5179
; READ BACK THE VIA'S REGISTERS
5180
5181
5182 025230 012703 000020 MOV #PATCM-PATCR,R3 ;GET COUNT OF # OF REG'S TO BE READ
5183 025234 012737 120000 025254 MOV #ORB,7$ ;ADDRESS OF FIRST REGISTER
5184 025242 012737 003122 025256 MOV #BT1,8$ ;DESTINATION BUFFER AREA
5185
5186 025250 004537 004064 6$: JSR R5,READ ;READ ONE REGISTER
5187 025254 000000 7$: 0 ;*** MODIFIED FROM ABOVE *** SOURCE ADDRESS
5188 025256 000000 8$: 0 ;*** MODIFIED IN LINE *** DESTINATION ADDRESS
5189 025260 103002 BCC 33$ ;IF AN ERROR OCCURED,
5190 025262 ERROR ;REPORT IT &
5191 025262 104460 TRAP C$ERROR
5192 025264 000466 BR 25$ ; EXIT
5193
5194 025266 005237 025254 33$: INC 7$ ;POINT TO NEXT REGISTER
5195 025272 005237 025256 INC 8$ ;POINT TO NEXT BUFFER LOCATION
5196 025276 077314 SOB R3,6$ ;LOOP UNTIL ALL REGISTERS HAVE BEEN READ
5197
; CHECK THE VALUES READ AGAINST THE EXPECTED VALUES
5198
5199
5200 025300 012701 002604 MOV #PATCR,R1 ;POINTER TO EXPECTED DATA VALUES
5201 025304 012702 003122 MOV #BT1,R2 ;POINTER TO DATA READ
5202 025310 012704 003206 MOV #BT2,R4 ;POINTER TO 'XOR' VALUES
5203 025314 012705 002624 MOV #PATCM,R5 ;POINTER TO 'MASK' VALUES
5204 025320 012703 000010 MOV #8.,R3 ;NUMBER OF WORDS TO BE PROCESSED
5205 025324 005037 002332 CLR ERRFLG ;RESET THE ERROR FLAG
5206
5207 025330 012114 9$: MOV (R1)+,(R4) ;GET EXPECTED VALUE (2 BYTES AT A TIME)
5208 025332 012200 MOV (R2)+,R0 ;GET ACTUAL VALUE AND SETUP FOR 'XOR'
5209 025334 074014 XOR R0,(R4) ;DEVELOPE 'XOR'
5210 025336 042524 BIC (R5)+,(R4)+ ;CLEAR THOSE BITS WE DON'T CARE ABOUT
5211 025340 001402 BEQ 10$ ;IF NO ERROR, SKIP NEXT INSTRUCTION
5212 025342 005237 002332 INC ERRFLG ;IF ERROR, SET FLAG TO SAY SO!
5213 025346 077310 10$: SOB R3,9$ ;LOOP UNTIL ALL VALUES CHECKED
5214
5215 025350 005737 002332 TST ERRFLG ;WAS THERE AN ERROR DETECTED?
5216 025354 001406 BEQ 12$ ;NO, PROCEED WITH TESTING
5217 025356 GEDF EM20,ERR6 ;YES, REPORT A VIA REGISTER ERROR
5218 ; 'DEVICE FATAL' ERROR # 27
5219 025356 104455 TRAP C$ERDF
5220 025360 000033 .WORD 27

```

CVDMAA.P11 12-DEC-80 15:59

TEST 11 -- VIA REGISTER ADDRESSING

```

5221 025362 015243
5222 025364 005650
5223 025366          ESCAPE TST          ;EXIT FROM THIS TEST -- LOOP IF REQUESTED
5224 025366 104410          ;TRAP C$ESCAPE
5225 025370 000344          ;WORD L10041-.
5226
5227
5228
5229 025372 004537 004176 12$: JSR R5,READI ;GET THE CURRENT VALUE OF THE VIA'S
5230 025376 120003          ; 'DDRA' REGISTER FOR LATER ERROR CHECKING
5231 025400 000000          15$: 0
5232 025402 103002          BCC 34$ ;IF AN ERROR OCCURED,
5233 025404          ERROR ;REPORT IT &
5234 025404 104460          TRAP C$ERROR
5235 025406 000415          BR 25$ ; EXIT
5236 025410 004537 004322 34$: JSR R5,WRITEI ;LOAD DDRB WITH 000
5237 025414 120002          DDRB
5238 025416 000000          0
5239 025420 103002          BCC 35$ ;IF AN ERROR OCCURED,
5240 025422          ERROR ;REPORT IT &
5241 025422 104460          TRAP C$ERROR
5242 025424 000406          BR 25$ ; EXIT
5243 025426 004537 004064 35$: JSR R5,READ ;READ IT BACK AND CHECK IT
5244 025432 120002          DDRB
5245 025434 002312          BDATA
5246 025436 103002          BCC 36$ ;IF AN ERROR OCCURED,
5247 025440          ERROR ;REPORT IT &
5248 025440 104460          TRAP C$ERROR
5249 025442 000534          BR 24$ ; EXIT
5250 025444 105737 002312 36$: TSTB BDATA ;THIS SHOULD NOW BE ZERO
5251 025450 001413          BEQ 14$ ;IT IS, PRECEDE TESTING
5252 025452 105037 002310 CLRB GDATA ;IT ISN'T! SETUP FOR & REPORT ERROR
5253 025456 012737 000002 002334 MOV #2,REGNUM ;IDENTIFY THE DDRB REG.
5254 025464          GEDF EM21,ERR7 ;REPORT ERROR
5255          ; 'DEVICE FATAL' ERROR # 28
5256 025464 104455          TRAP C$ERDF
5257 025466 000034          .WORD 28
5258 025470 015442          .WORD EM21
5259 025472 006612          .WORD ERR7
5260 025474          ESCAPE TST          ;EXIT FROM THIS TEST -- LOOP IF REQUESTED
5261 025474 104410          ;TRAP C$ESCAPE
5262 025476 000236          ;WORD L10041-.
5263
5264 025500 113737 025400 002310 14$: MOVB 15$,GDATA ;THIS IS WHAT WE EXPECT TO READ NOW
5265 025506 004537 004064 JSR R5,READ ;READ BACK DDRA -- IT SHOULD BE = 366
5266 025512 120003          DDRA
5267 025514 002312          BDATA
5268 025516 103002          BCC 37$ ;IF AN ERROR OCCURED,
5269 025520          ERROR ;REPORT IT &
5270 025520 104460          TRAP C$ERROR
5271 025522 000504          BR 24$ ; EXIT
5272 025524 123737 002310 002312 37$: CMPB GDATA,BDATA ;IS IT REALLY A 377?
5273 025532 001411          BF^ 16$ ;YES, PROCEED WITH TESTING
5274 025534 012737 000003 002334 MUV #3,REGNUM ;IDENTIFY THE DDRA REG.
5275 025542          GEDF EM22,ERR7 ;NO, REPORT ERROR
5276          ; 'DEVICE FATAL' ERROR # 29

```



CVDMAA.P11 12-DEC-80 15:59

TEST 11 -- VIA REGISTER ADDRESSING

```

5277 025542 104455
5278 025544 000035
5279 025546 015477
5280 025550 006612
5281 025552
5282 025552 104410
5283 025554 000160
5284
5285 025556 004537 004322 16$: JSR R5,WRITEI ;RE-LOAD DDRB WITH 377
5286 025562 120002
5287 025564 177777 17$: DDRB -1
5288 025566 103002 BCC 38$ ;IF AN ERROR OCCURED,
5289 025570 ERROR ;REPORT IT &
5290 025570 104460 TRAP C$ERROR
5291 025572 000460
5292 025574 004537 004322 38$: BR 24$ ; EXIT
5293 025600 120003 JSR R5,WRITEI ;AND NOW CLEAR DDRA TO ZEROS
5294 025602 000000 DDRA 0
5295 025604 103002 BCC 39$ ;IF AN ERROR OCCURED,
5296 025606 ERROR ;REPORT IT &
5297 025606 104460 TRAP C$ERROR
5298 025610 000451 BR 24$ ; EXIT
5299
5300 025612 004537 004064 39$: JSR R5,READ ;NOW, DID DDRA GO TO ZEROES
5301 025616 120003 DDRA
5302 025620 002312 BDATA
5303 025622 105737 002312 TSTB BDATA
5304 025626 001413 BEQ 18$ ;YES, BUT WHAT ABOUT DDRB?
5305 025630 105037 002310 CLRB GDATA ;NO, SETUP FOR AND
5306 025634 012737 000003 002334 MOV #3,REGNUM ;IDENTIFY THE DDRA REG.
5307 025642 GEDF EM21,ERR7 ; REPORT THE ERROR
5308 ; 'DEVICE FATAL' ERROR # 30
5309 025642 104455 TRAP C$ERDF
5310 025644 000036 .WORD 30
5311 025646 015442 .WORD EM21
5312 025650 006612 .WORD ERR7
5313 025652
5314 025652 104410 ESCAPE TS1 ;EXIT FROM THIS TEST -- LOOP IF REQUESTED
5315 025654 000060 TRAP C$ESCAPE
5316 .WORD L10041-.
5317 025656 004537 004064 18$: JSR R5,READ ;WHAT ABOUT DDRB -- IT SHOULD BE 377 NOW
5318 025662 120002 DDRB
5319 025664 002312 BDATA
5320 025666 103002 BCC 40$ ;IF AN ERROR OCCURED,
5321 025670 ERROR ;REPORT IT &
5322 025670 104460 TRAP C$ERROR
5323 025672 000420 BR 24$ ; EXIT
5324 025674 123737 002312 025564 40$: CMPB BDATA,17$ ;IS IT?
5325 025702 001414 BEQ 24$ ;YES, EXIT TEST
5326 025704 113737 025564 002310 MOVB 17$,GDATA ;NO, SETUP FOR AND
5327 025712 012737 000002 002334 MOV #2,REGNUM ;IDENTIFY THE DDRB REG.
5328 025720 GEDF EM22A,ERR7 ; REPORT ERROR
5329 ; 'DEVICE FATAL' ERROR # 31
5330 025720 104455 TRAP C$ERDF
5331 025722 000037 .WORD 31
5332 025724 015532 .WORD EM22A

```

CVDMAA.P11 12-DEC-80 15:59

TEST 11 -- VIA REGISTER ADRESSING

5333	025726	006612
5334	025730	
5335	025730	104410
5336	025732	000002
5337		
5338	025734	
5339	025734	
5340	025734	104401

ESCAPE TST

;EXIT FROM THIS TEST -- LOOP IF .WORD ERR7  
 REQUESTED  
 TRAP C\$ESCAPE  
 .WORD L10041-

24\$: ENDTST

L10041:  
 TRAP C\$ETST



CVDMAA.P11 12-DEC-80 15:59

TEST 12 -- VIA'S DDRB DATA READ/WRITE

.SBTTL TEST 12 -- VIA'S DDRB DATA READ/WRITE

```

:*****
:*
:*      TEST 12 -- VIA'S DDRB DATA READ/WRITE
:*
:*      DDRB == 'DATA DIRECTION REGISTER B'
:* FIRST, A MASTER CLEAR IS PERFORMED. THEN :
:* READ/WRITE BITS 0-7 OF VIA DATA DIRECTION REGISTER B ARE TESTED BY WRITING,
:* READING, AND COMPARING EACH BYTE OF DATA PATTERN B.
:* DATA PATTERN B = 125, 252, 000, 377, 001, 002, 004, 010, 020, 040, 100,
:*                   200, 376, 375, 373, 367, 357, 337, 277, 177, 000
:*****

```

```

5341
5342
5343
5344
5345
5346
5347
5348
5349
5350
5351
5352
5353
5354
5355
5356
5357 025736
5358 025736 004737 003762
5359 025742 103003
5360 025744
5361 025744 104460
5362 025746
5363 025746 104410
5364 025750 000046
5365
5366 025752 012701 002526
5367 025756 012103
5368
5369 025760
5370 025760
5371 025760
5372 025760 104402
5373
5374 025762 111137 002306
5375 025766 112137 002310
5376 025772 012700 120002
5377 025776 004737 005034
5378 026002 103003
5379 026004
5380 026004 104460
5381 026006
5382 026006 104410
5383 026010 000006
5384
5385 026012
5386 026012
5387 026012 104403
5388
5389 026014 077317
5390
5391
5392 026016
5393 026016
5394 026016 104401

```

```

:
:      BGNTST
:
:      JSR      PC,MSTCLR      ;INIT DMV & START UP THE MAINT. LOOP
:      BCC      30$           ;IF AN ERROR OCCURED,
:      ERROR    ;REPORT IT &
:
:      ESCAPE   TST           ; EXIT
:
:      TRAP     C$ERROR
:
:      TRAP     C$ESCAPE
:      .WORD    L10042-.
:
:      30$:     MOV      #PATB,R1 ;POINT TO PATTERN TABLE
:      MOV      (R1)+,R3      ;GET # OF ENTRIES IN TABLE
:
:      T12.LP:  BGNSUB        ;THE SUBTEST ONLY TESTS THE ONE PATTERN
:
:      TRAP     C$SUBSUB
:      T12.1:
:
:      MOVB     (R1),TDATA    ;SETUP TEST DATA BYTE FOR 'STREG'
:      MOVB     (R1)+,GDATA   ;SETUP EXPECTED DATA BYTE FOR 'STREG'
:      MOV      #DDRb,R0     ;SPECIFY THE REGISTER BEING TESTED
:      JSR      PC,STREG     ;PERFORM STATIC TEST OF THE SPECIFIED REGISTER
:      BCC      10$          ;WAS AN ERROR FOUND?
:      ERROR    ;YES, REPORT IT AND
:
:      ESCAPE   TST           ; EXIT FROM THE TEST. 'CKLOOP' IS IMPLIED
:
:      TRAP     C$ERROR
:      TRAP     C$ESCAPE
:      .WORD    L10042-.
:
:      10$:     ENDSUB
:
:      L10043:  TRAP     C$ESUB
:
:      SOB      R3,T12.LP    ;IF THERE IS IN FACT MORE DATA, LOOP BACK TO
:      ;TEST IT. ELSE, FALL OUT OF LOOP AND TEST
:
:      ENDTST
:
:      L10042:  TRAP     C$ETST

```

CVDMAA.P11 12-DEC-80 15:59

TEST 13 -- VIA'S DDRA DATA READ/WRITE

.SBTTL TEST 13 -- VIA'S DDRA DATA READ/WRITE

```

+++++
*
* TEST 13 -- VIA'S DDRA DATA READ/WRITE
*
* DDRA == 'DATA DIRECTION REGISTER A'
*
* FIRST, A MASTER CLEAR IS PERFORMED. THEN :
* READ/WRITE BITS 0-7 OF VIA DATA DIRECTION REGISTER A ARE TESTED BY WRITING,
* READING, AND COMPARING EACH BYTE OF DATA PATTERN B.
* DATA PATTERN B = 125, 252, 000, 377, 001, 002, 004, 010, 020, 040, 100,
*                   200, 376, 375, 373, 367, 357, 337, 277, 177, 000
-----

```

```

5395
5396
5397
5398
5399
5400
5401
5402
5403
5404
5405
5406
5407
5408
5409
5410
5411
5412 026020
5413 026020 004737 003762
5414 026024 103003
5415 026026
5416 026026 104460
5417 026030
5418 026030 104410
5419 026032 000046
5420
5421 026034 012701 002526
5422 026040 012103
5423
5424 026042
5425 026042
5426 026042
5427 026042 104402
5428
5429 026044 111137 002306
5430 026050 112137 002310
5431 026054 012700 120003
5432 026060 004737 005034
5433 026064 103003
5434 026066
5435 026066 104460
5436 026070
5437 026070 104410
5438 026072 000006
5439
5440 026074
5441 026074
5442 026074 104403
5443
5444 026076 077317
5445
5446
5447 026100
5448 026100
5449 026100 104401

:      BGNTST
:
:      JSR      PC,MSTCLR      ;INIT DMV & START UP THE MAINT. LOOP
:      BCC      30$           ;IF AN ERROR OCCURED,
:      ERROR                                ;REPORT IT &
:
:      ESCAPE  TST           ; EXIT
:
:
:      30$:     MOV      #PATB,R1      ;POINT TO PATTERN TABLE
:              MOV      (R1)+,R3      ;GET # OF ENTRIES IN TABLE
:
:      T13.LP:  BGNSUB          ;THE SUBTEST ONLY TESTS THE ONE PATTERN
:
:
:              T13.1:          TRAP    C$SUB
:
:      MOVB     (R1),TDATA      ;SETUP TEST DATA BYTE FOR 'STREG'
:      MOVB     (R1)+,GDATA     ;SETUP EXPECTED DATA BYTE FOR 'STREG'
:      MOV      #DDRA,R0        ;SPECIFY THE REGISTER BEING TESTED
:      JSR      PC,STREG        ;PERFORM STATIC TEST OF THE SPECIFIED REGISTER
:      BCC      10$           ;WAS AN ERROR FOUND?
:      ERROR                                ;YES, REPORT IT AND
:
:      ESCAPE  TST           ; EXIT FROM THE TEST. 'CKLOOP' IS IMPLIED
:
:
:              TRAP    C$ERROR
:              TRAP    C$ESCAPE
:              .WORD   L10044-
:
:      10$:    ENDSUB
:
:
:              L10045:        TRAP    C$ESUB
:
:      SOB      R3,T13.LP      ;IF THERE IS IN FACT MORE DATA, LOOP BACK TO
:
:
:      ENDTST
:
:              L10044:        TRAP    C$SETST

```



CVDMAA.P11 12-DEC-80 15:59

TEST 14 -- VIA'S ORB DATA READ/WRITE

.SBTTL TEST 14 -- VIA'S ORB DATA READ/WRITE

```

:*****
:*
:* TEST 14 -- VIA'S ORB DATA READ/WRITE
:* ORB == 'OUTPUT REGISTER PORT B'
:*
:* FIRST, A MASTER CLEAR IS PERFORMED. NEXT, 377 IS LOADED INTO DATA
:* DIR. REG. B (DDRB) TO SET ALL B PORT PINS FOR OUTPUT MODE. THEN
:* READ/WRITE BITS 0-7 OF VIA OUTPUT REG. PORT B ARE TESTED BY WRITING,
:* READING, AND COMPARING EACH BYTE OF DATA PATTERN B.
:* DATA PATTERN B = 125, 252, 000, 377, 001, 002, 004, 010, 020, 040, 100,
:*                   200, 376, 375, 373, 367, 357, 337, 277, 177, 000
:*****

```

5450  
5451  
5452  
5453  
5454  
5455  
5456  
5457  
5458  
5459  
5460  
5461  
5462  
5463  
5464  
5465  
5466  
5467  
5468  
5469  
5470  
5471  
5472  
5473  
5474  
5475  
5476  
5477  
5478  
5479  
5480  
5481  
5482  
5483  
5484  
5485  
5486  
5487  
5488  
5489  
5490  
5491  
5492  
5493  
5494  
5495  
5496  
5497  
5498  
5499  
5500  
5501  
5502  
5503  
5504  
5505

```

026102
026102 004737 003762
026106 103003
026110 104460
026112
026112 104410
026114 000066
026116 004537 004322
026122 120002
026124 177777
026126 103003
026130 104460
026132 104410
026134 000046
026136 012701 002526
026142 012103
026144
026144 104402
026146 111137 002306
026152 112137 002310
026156 012700 120000
026162 004737 005034
026166 103003
026170 104460
026172 104410
026174 000006

```

```

: BGNTST
:
: T14::
: JSR PC,MSTCLR ;INIT DMV & START UP THE MAINT. LOOP
: BCC 30$ ;IF AN ERROR OCCURED,
: ERROR ;REPORT IT &
: ESCAPE TST ; EXIT TRAP C$ERROR
: TRAP C$ESCAPE
: .WORD L10046-.
30$: JSR R5,WRITEI ;INITIALIZE PORT B FOR I/O
: DDRB
: -1
: BCC 31$ ;IF AN ERROR OCCURED,
: ERROR ;REPORT IT &
: ESCAPE TST ; EXIT TRAP C$ERROR
: TRAP C$ESCAPE
: .WORD L10046-.
31$: MOV #PATB,R1 ;POINT TO PATTERN TABLE
: MOV (R1)+,R3 ;GET # OF ENTRIES IN TABLE
T14.LP: BGNSUB ;THE SUBTEST ONLY TESTS THE ONE PATTERN
: T14.1: TRAP C$BSUB
: MOVB (R1),TDATA ;SETUP TEST DATA BYTE FOR 'STREG'
: MOVB (R1)+,GDATA ;SETUP EXPECTED DATA BYTE FOR 'STREG'
: MOV #ORB,R0 ;SPECIFY THE REGISTER BEING TESTED
: JSR PC,STREG ;PERFORM STATIC TEST OF THE SPECIFIED REGISTER
: BCC 10$ ;WAS AN ERROR FOUND?
: ERROR ;YES, REPORT IT AND
: ESCAPE TST ; EXIT FROM THE TEST. 'CKLOOP' IS IMPLIED
: TRAP C$ERROR
: TRAP C$ESCAPE
: .WORD L10046-.

```

CVDMAA.P11 12-DEC-80 15:59

TEST 14 -- VIA'S ORB DATA READ/WRITE

5506 026176  
 5507 026176  
 5508 026176 104403  
 5509  
 5510 026200 077317  
 5511  
 5512  
 5513 026202  
 5514 026202  
 5515 026202 104401

10\$: ENDSUB

L10047: TRAP C\$ESUB

SOB R3,T14.LP

:IF THERE IS IN FACT MORE DATA, LOOP BACK TO  
:TEST IT. ELSE, FALL OUT OF LOOP AND TEST

ENDTST

L10046: TRAP C\$ETST



CVDMAA.P11

12-DEC-80 15:59

TEST 15 -- VIA'S T1 DATA READ/WRITE

.SBTTL TEST 15 -- VIA'S T1 DATA READ/WRITE

5516  
5517  
5518  
5519  
5520  
5521  
5522  
5523  
5524  
5525  
5526  
5527  
5528  
5529  
5530  
5531  
5532  
5533  
5534  
5535  
5536  
5537  
5538  
5539  
5540  
5541  
5542  
5543  
5544  
5545  
5546  
5547  
5548  
5549  
5550  
5551  
5552  
5553  
5554  
5555  
5556  
5557  
5558  
5559  
5560  
5561  
5562  
5563  
5564  
5565  
5566  
5567  
5568  
5569  
5570  
5571

```

:*****
:*
:*   TEST 15 -- VIA'S T1 DATA READ/WRITE
:*
:*   T1 == 'TIMER #1'
:*
:* THIS TEST WRITES, READS, AND CHECKS THE T1 LATCHES AND COUNTER REGISTERS
:* WITH DATA PATTERNS IN EACH OF 3 SUBTESTS.
:*
:* FIRST SUBTEST: CHECKS FOR PROPER LOADING OF LATCHES
:* IT ALSO CHECKS TO BE SURE THAT THE COUNTER APPEARS TO BE DOING
:* SOMETHING TO THE COUNTERS. AS LONG AS THEY HAVE CHANGED FROM THE
:* VALUE LOADED INTO THEM, WE WILL BE SATISFIED.
:*
:* A. A MASTER CLEAR IS PERFORMED.
:* B. ALL LATCHES ARE LOADED TO ZEROES (JUST IN CASE), ACR6 & ACR7 ARE SET
:* TO ZERO (MODE 00), AND 'T1' INTERRUPT ENABLE FLAG IS CLEARED.
:*
:* C. T1L-L(ADR 04) IS LOADED WITH THE CURRENT BYTE OF DATA PATTERN B.
:* D. T1L-L(ADR 06) IS READ AND COMPARED TO THE BYTE JUST WRITTEN.
:* E. T1C-L(ADR 04) IS READ AND CHECKED TO BE DIFFERENT THAN THE TEST BYTE.
:*
:* F. T1L-L(ADR 06) IS LOADED WITH THE COMPLEMENT OF THE CURRENT DATA BYTE.
:* G. T1L-L(ADR 06) IS READ AND COMPARED TO THE BYTE JUST WRITTEN.
:*
:* H. T1L-L(ADR 06) IS RE-LOADED WITH 0 TO MAKE T1C-H DECREMENT FAST.
:*   T1L-H(ADR 05) IS LOADED WITH THE ORIGINAL TEST DATA PATTERN BYTE.
:* I. T1L-H(ADR 07) IS READ AND COMPARED TO THE BYTE LOADED INTO T1L-H.
:*
:* J. T1C-H(ADR 05) IS READ AND CHECKED TO BE DIFFERENT THAN THE TEST BYTE.
:*
:* K. T1L-H(ADR 07) IS LOADED WITH THE COMPLEMENT OF THE CURRENT DATA BYTE.
:* L. T1L-H(ADR 07) IS READ AND COMPARED TO THE BYTE JUST LOADED.
:*
:* M. STEPS C-L ARE REPEATED USING EACH BYTE OF DATA PATTERN B.
:*
:* SECOND SUBTEST: CHECKS FOR CROSS-TALK AND ADDRESSING ERRORS
:* FROM T1L-L TO T1L-H
:*
:* A. T1L-H(ADR 07) IS LOADED WITH 000 TO CLEAR IT.
:* B. T1L-L(ADR 06) IS LOADED WITH A BYTE OF DATA PATTERN B.
:* C. T1L-L(ADR 06) IS READ AND COMPARED TO THE DATA JUST WRITTEN.
:* D. T1L-H(ADR 07) IS READ AND COMPARED TO 000.
:* E. STEPS B-D ARE REPEATED USING EACH BYTE OF DATA PATTERN B.
:*
:* THIRD SUBTEST: CHECKS FOR CROSS-TALK AND ADDRESSING ERRORS
:* FROM T1L-H TO T1L-L
:*
:* A. T1L-L(ADR 04) IS LOADED WITH 000 TO CLEAR IT
:* B. T1L-H(ADR 07) IS LOADED WITH A BYTE OF DATA PATTERN B.
:* C. T1L-H(ADR 07) IS READ AND COMPARED TO THE DATA JUST WRITTEN.

```

CVDMAA.P11 12-DEC-80 15:59

TEST 15 -- VIA'S T1 DATA READ/WRITE

5572  
5573  
5574  
5575  
5576  
5577  
5578  
5579  
5580  
5581 026204  
5582  
5583  
5584  
5585 026204 004737 003762  
5586 026210 103003  
5587 026212  
5588 026212 104460  
5589 026214  
5590 026214 104410  
5591 026216 001030  
5592 026220  
5593  
5594  
5595  
5596 026220 004537 004660  
5597 026224 000000  
5598 026226 000000  
5599 026230 103003  
5600 026232  
5601 026232 104460  
5602 026234  
5603 026234 104410  
5604 026236 001010  
5605  
5606  
5607  
5608  
5609  
5610  
5611  
5612 026240  
5613 026240  
5614 026240 104402  
5615 026242 012701 002526  
5616 026246 012103  
5617  
5618 026250  
5619 026250 112137 002306  
5620 026254 013737 002306 002310  
5621  
5622 026262  
5623 026262 104404  
5624  
5625  
5626  
5627 026264 004537 004310

```

;* D. T1L-L(ADR 06) IS READ AND COMPARED TO 000.
;* E. STEPS B-D ARE REPEATED USING EACH BYTE OF DATA PATTERN B.
;*
;* DATA PATTERN B = 125, 252, 000, 377, 001, 002, 004, 010, 020, 040, 100,
;*                   200, 376, 375, 373, 367, 357, 337, 277, 177, 000
;*
;-----
;
;          BGNTST
;
;          T15::
;
; ***** STEP A *****
;
;          JSR    PC,MSTCLR      ;INIT DMV & START UP M-LOOP
;          BCC    1$             ;IF NO ERRORS, PROCEED
;          ERROR  ;ELSE, REPORT ERROR &
;
;          ESCAPE TST           ; GET OUT OF THE TEST          TRAP    C$ERROR
;
;          1$:
;
;          .WORD  L10050-
;
; ***** STEP B *****
;
;          JSR    R5,INITT1     ;INITIALIZE THE TIMER'S REGISTERS
;          0
;          .WORD  0             ; WITH ZEROES
;          BCC    .+10         ; 00 --> ACR6 & ACR7 AND DISABLE INTERRUPTS
;          ERROR  ;ELSE, REPORT IT
;
;          ESCAPE TST           ; AND EXIT THIS TEST          TRAP    C$ERROR
;
;          .WORD  L10050-
;
;          ;WE WANT THE LEAST ACTIVE OPERATING MODE FOR THIS TIMER WHILE WE ARE TESTING
;          ;IT. THE MODE WE'RE USING HERE IS DOCUMENTED THUSLY: 'GENERATE A SINGLE
;          ;TIME-OUT INTERRUPT EACH TIME T1 IS LOADED. PB7 DISABLED.'
;          ;AS AN ADDED PRECAUTION, WE ARE DISABLING INTERRUPTS BY CLEARING THE 'T1' FLAG
;          ;WITHIN 'IER'.
;
;          BGNSUB               ;BEGIN THE FIRST SUBTEST
;
;          T15.1:
;          TRAP    C$BSUB
;
;          MOV    #PATB,R1      ;POINT TO THE APPROPRIATE PATTERN TABLE
;          MOV    (R1)+,R3      ;EXTRACT THE BYTE COUNT FROM THE TABLE
;
;          T16.LP:
;          MOVB   (R1)+,JDATA    ;GET ONE BYTE OF THE TEST DATA
;          MOV    TDATA,GDATA    ;THE TEST DATA IS NORMALLY THE GOOD DATA TOO
;
;          BGNSEG
;
;          TRAP    C$BSEG
;
; ***** STEP C *****
;
;          JSR    R5,WRITE      ;LOAD T1L-L(ADDR 04)

```



CVDMAA.P11 12-DEC-80 15:59

TEST 15 -- VIA'S T1 DATA READ/WRITE

```

5628 026270 120004          T1CL
5629 026272 002306          TDATA          ;THE TEST DATA FROM 'TDATA'
5630
5631
5632          ; ****-----*** STEP D ****-----***
5633
5634 026274 004537 004064      JSR      R5,READ          ;READ T1L-L(ADDR 06)
5635 026300 120006          T1LL
5636 026302 002312          BDATA
5637 026304 123737 002310 002312  CMPB     GDATA,BDATA      ;AND CHECK IT
5638 026312 001407          BEQ      2$              ;IF OK, PROCEED
5639 026314 012737 000006 002334  MOV      #6,REGNUM        ;IDENTIFY THE FAILING REGISTER &
5640 026322          GEDF     EM20,ERR7        ; REPORT FAILURE
5641          ;          'DEVICE FATAL' ERROR # 32
5642 026322 104455          TRAP    CSERDF
5643 026324 000040          .WORD   32
5644 026326 015243          .WORD   EM20
5645 026330 006612          .WORD   ERR7
5646
5647          ; ****-----*** STEP E ****-----***
5648
5649 026332 004537 004064      2$:     JSR      R5,READ          ;READ T1C-L(ADDR 04)
5650 026336 120004          T1CL
5651 026340 002312          BDATA
5652 026342 123737 002310 002312  CMPB     GDATA,BDATA      ;AND CHECK IT. SEEING AS THE TIMER IS RUNNING,
5653 026350 001017          BNE     4$              ;THIS MUST NOT EQUAL THE SET VALUE!
5654 026352 004537 004064      JSR      R5,READ          ;IF IT IS, MAYBE WE JUST READ IT AT THE WRONG
5655 026356 120004          T1CL                    ;TIME! RE-READ AND CHECK ONE MORE TIME.
5656 026360 002312          BDATA
5657 026362 123737 002310 002312  CMPB     GDATA,BDATA      ;CHECK IT AGAIN, SAM.
5658 026370 001007          BNE     4$              ;THIS TIME IT SHOULD BE DIFFERENT.
5659          ;OTHERWISE, WE HAVE A LEGITIMATE FAILURE
5660 026372 012737 000004 002334  MOV      #4,REGNUM        ; IDENTIFY THE FAILING REGISTER &
5661 026400          GEDF     EM20A,ERR7      ; REPORT FAILURE
5662          ;          'DEVICE FATAL' ERROR # 33
5663 026400 104455          TRAP    CSERDF
5664 026402 000041          .WORD   33
5665 026404 015275          .WORD   EM20A
5666 026406 006612          .WORD   ERR7
5667
5668          ; ****-----*** STEP F ****-----***
5669
5670 026410 105137 002306      4$:     COMB     TDATA          ;USE THE ONE'S COMPLEMENT THIS TIME
5671 026414 105137 002310      COMB     GDATA          ;THE EXPECTED DATA IS ALSO THE COMPLEMENT
5672 026420 004537 004310      JSR      R5,WRITE        ;LOAD T1L-L(ADDR 06)
5673 026424 120006          T1LL
5674 026426 002306          TDATA          ;THE TEST DATA FROM 'TDATA'
5675
5676          ; ****-----*** STEP G ****-----***
5677
5678 026430 004537 004064      6$:     JSR      R5,READ          ;READ T1L-L(ADDR 06)
5679 026434 120006          T1LL
5680 026436 002312          BDATA
5681 026440 123737 002310 002312  CMPB     GDATA,BDATA      ;AND CHECK IT
5682 026446 001407          BEQ      8$              ;IF OK, PROCEED
5683 026450 012737 000006 002334  MOV      #6,REGNUM        ;IDENTIFY THE FAILING REGISTER &

```

CVDMAA.P11 12-DEC-80 15:59

TEST 15 -- VIA'S T1 DATA READ/WRITE

```

5684 026456          GEDF  EM20,ERR7      ; REPORT FAILURE
5685                                     ; 'DEVICE FATAL' ERROR # 34
5686 026456 104455                                     TRAP  C$ERDF
5687 026460 000042                                     .WORD 34
5688 026462 015243                                     .WORD EM20
5689 026464 006612                                     .WORD ERR7
5690
5691 ; ****--*** STEP H ****--***
5692
5693 026466 105137 002306 8$:  COMB  TDATA      ;RESTORE THE DATA TO THE ORIGINAL VALUE
5694 026472 105137 002310      COMB  GDATA
5695 026476 004537 004322      JSR   R5,WRITEI    ;SET THE LOW LATCH TO MAKE SURE THE HIGH
5696 026502 120006              T1LL              ;COUNTER IS DOING MOST OF THE WORK
5697 026504 000001              1
5698 026506 004537 004310      JSR   R5,WRITE    ;LOAD T1L-H(ADDR 05)
5699 026512 120005              T1CH
5700 026514 002306              TDATA            ;THE TEST DATA FROM 'TDATA'
5701
5702 ; ****--*** STEP I ****--***
5703
5704 026516 004537 004064      JSR   R5,READ      ;READ T1L-H(ADDR 07)
5705 026522 120007              T1LH
5706 026524 002312              BDATA
5707 026526 123737 002310 002312  CMPB  GDATA,BDATA  ;AND CHECK IT
5708 026534 001407              BEQ   10$          ;IF OK, PROCEED
5709 026536 012737 000007 002334  MOV   #7,REGNUM    ;IDENTIFY THE FAILING REGISTER &
5710 026544          GEDF  EM20,ERR7      ; REPORT FAILURE
5711                                     ; 'DEVICE FATAL' ERROR # 35
5712 026544 104455                                     TRAP  C$ERDF
5713 026546 000043                                     .WORD 35
5714 026550 015243                                     .WORD EM20
5715 026552 006612                                     .WORD ERR7
5716
5717 ; ****--*** STEP J ****--***
5718
5719 026554 004537 004064 10$:  JSR   R5,READ      ;READ T1C-H(ADDR 05)
5720 026560 120005              T1CH
5721 026562 002312              BDATA
5722 026564 012737 000005 002334  MOV   #5,REGNUM    ;IDENTIFY THE REGISTER BEING CHECKED
5723 026572 105737 002306              TSTB  TDATA        ;WAS THE TEST DATA '000'?
5724 026576 001410              BEQ   14$          ;YES, THEN WE CAN'T BE SURE OF THE RESULTS!
5725 026600 123737 002310 002312  CMPB  GDATA,BDATA  ;NO, CHECK IT
5726 026606 001004              BNE   14$          ;IT SHOULDN'T = THE LOADED VALUE
5727 026610          GEDF  EM20A,ERR7    ;IT DID! REPORT FAILURE
5728                                     ; 'DEVICE FATAL' ERROR # 36
5729 026610 104455                                     TRAP  C$ERDF
5730 026612 000044                                     .WORD 36
5731 026614 015275                                     .WORD EM20A
5732 026616 006612                                     .WORD ERR7
5733
5734 ; ****--*** STEP K ****--***
5735
5736 026620 105137 002306 14$:  COMB  TDATA      ;USE THE ONE'S COMPLEMENT THIS TIME
5737 026624 105137 002310      COMB  GDATA
5738 026630 004537 004310      JSR   R5,WRITE    ;THE EXPECTED DATA IS ALSO THE COMPLEMENT
5739 026634 120007              T1LH            ;LOAD T1L-H(ADDR 07)

```



CVDMAA.P11 12-DEC-80 15:59

TEST 15 -- VIA'S T1 DATA READ/WRITE

```

5740 026636 002306          TDATA          ;THE TEST DATA FROM 'TDATA'
5741
5742
5743 ; ****----- STEP L ****-----
5744 026640 004537 004064      JSR      R5,READ          ;READ T1L-H(ADDR 07)
5745 026644 120007              T1LH
5746 026646 002312              BDATA
5747 026650 123737 002310 002312  CMPB     GDATA,BDATA     ;AND CHECK IT
5748 026656 001407              BEQ      16$              ;IF OK, PROCEED
5749 026660 012737 000007 002334  MOV      #7,REGNUM        ;IDENTIFY THE FAILING REGISTER &
5750 026666              GDF      EM20,ERR7        ; REPORT FAILURE
5751 ;                               ; 'DEVICE FATAL' ERROR # 37
5752 026666 104455              TRAP    C$ERDF
5753 026670 000045              .WORD  37
5754 026672 015243              .WORD  EM20
5755 026674 006612              .WORD  ERR7
5756
5757 ; ****----- STEP M ****-----
5758
5759 026676          16$:      ENDSEG
5760 026676
5761 026676 104405          10000$: TRAP    C$ESEG
5762
5763 026700 000402              BR      21$
5764 026702 000137 026250      20$:    JMP      T16.LP
5765 026706 077303          21$:    SOB     R3,20$
5766 ;                               ;IF MORE DATA, DO ANOTHER BYTE
5767 026710          ENDSUB          ;ELSE, EXIT SUBTEST
5768 026710
5769 026710 104403          L10051: TRAP    C$ESUB
5770
5771
5772 026712          BGNSUB          ;BEGIN THE SECOND SUBTEST
5773 026712
5774 026712 104402          T15.2: TRAP    C$BSUB
5775 026714 012701 002526      MOV      #PATB,R1
5776 026720 012103          MOV      (R1)+,R3
5777 ;                               ;POINT TO THE APPROPRIATE PATTERN TABLE
5778 026722          T16.L1:
5779 026722 112137 002306          MOVB     (R1)+,TDATA
5780 026726 013737 002306 002310  MOV      TDATA,GDATA     ;THE TEST DATA IS NORMALLY THE GOOD DATA TOO
5781
5782
5783 ; ****----- STEP A ****-----
5784
5785 026734 004537 004322      JSR      R5,WRITEI        ;CLEAR T1L-H(ADDR 07)
5786 026740 120007              T1LH
5787 026742 000000              0          ;THE TEST DATA FROM 'TDATA'
5788
5789 026744          BGNSEG
5790 026744 104404          TRAP    C$BSEG
5791
5792 ; ****----- STEP B ****-----
5793
5794 026746 004537 004310      JSR      R5,WRITE
5795 026752 120006              T1LL          ;LOAD T1L-L(ADDR 06)

```

CVDMAA.P11 12-DEC-80 15:59

TEST 15 -- VIA'S T1 DATA READ/WRITE

```

5796 026754 002306          TDATA          ;THE TEST DATA FROM 'TDATA'
5797
5798
5799
5800 026756 004537 004064    JSR      R5,READ      ;READ T1L-L(ADDR 06)
5801 026762 120006
5802 026764 002312          BDATA
5803 026766 123737 002310 002312  CMPB    GDATA,BDATA  ;AND CHECK IT
5804 026774 001407          BEQ     2$            ;IF OK, PROCEED
5805 026776 012737 000006 002334  MOV     #6,REGNUM    ;IDENTIFY THE FAILING REGISTER &
5806 027004          GEDF    EM20,ERR7   ; REPORT FAILURE
5807
5808 027004 104455          ; 'DEVICE FATAL' ERROR # 38
5809 027006 000046          TRAP    C$ERDF
5810 027010 015243          .WORD  38
5811 027012 006612          .WORD  EM20
5812
5813
5814          ; ***** STEP D *****
5815 027014 004537 004064    2$:    JSR      R5,READ      ;READ T1L-H(ADDR 07)
5816 027020 120007
5817 027022 002312          BDATA
5818 027024 105737 002312    TSTB   BDATA        ;AND CHECK IT -- THIS SHOULD STILL BE ZERO
5819 027030 001411          BEQ     10$          ;IF OK, PROCEED
5820 027032 005037 002310    CLR    GDATA
5821 027036 012737 000007 002334  MOV     #7,REGNUM    ;IDENTIFY THE FAILING REGISTER &
5822 027044          GEDF    EM20B,ERR7  ; REPORT FAILURE
5823
5824 027044 104455          ; 'DEVICE FATAL' ERROR # 39
5825 027046 000047          TRAP    C$ERDF
5826 027050 015355          .WORD  39
5827 027052 006612          .WORD  EM20B
5828
5829          ; ***** STEP E *****
5830
5831 027054          10$:    ENDSEG
5832 027054
5833 027054 104405          10000$: TRAP    C$ESEG
5834
5835 027056 000402
5836 027060 000137 026722    20$:    BR      21$
5837 027064 077303          JMP     T16.L1
5838
5839 027066          21$:    SOB    R3,20$   ;IF MORE DATA, DO ANOTHER BYTE
5840 027066          ENDSUB          ;ELSE, EXIT SUBTEST
5841 027066 104403          L10052: TRAP    C$ESUB
5842
5843
5844 027070          BGNSUB          ;BEGIN THE THIRD SUBTEST
5845 027070
5846 027070 104402          T15.3: TRAP    C$BSUB
5847 027072 012701 002526    MOV     #PATB,R1    ;POINT TO THE APPROPRIATE PATTERN TABLE
5848 027076 012103          MOV     (R1)+,R3   ;EXTRACT THE BYTE COUNT FROM THE TABLE
5849
5850 027100          T16.L2:
5851 027100 112137 002306    MOVB   (R1)+,TDATA ;GET ONE BYTE OF THE TEST DATA

```



CVDMAA.P11 12-DEC-80 15:59

TEST 15 -- VIA'S T1 DATA READ/WRITE

```

5852 027104 013737 002306 002310      MOV      TDATA,GDATA      ;THE TEST DATA IS NORMALLY THE GOOD DATA TOO
5853
5854
5855      ; ***** STEP A *****
5856
5857 027112 004537 004322      JSR      R5,WRITEI      ;CLEAR T1L-L(ADDR 04)
5858 027116 120004
5859 027120 000000      T1CL
5860      0
5861 027122      BGNSEG
5862 027122 104404      TRAP      C$BSEG
5863
5864      ; ***** STEP B *****
5865
5866 027124 004537 004310      JSR      R5,WRITE      ;LOAD T1L-H(ADDR 07)
5867 027130 120007      T1LH
5868 027132 002306      TDATA      ;THE TEST DATA FROM 'TDATA'
5869
5870      ; ***** STEP C *****
5871
5872 027134 004537 004064      JSR      R5,READ      ;READ T1L-H(ADDR 07)
5873 027140 120007      T1LH
5874 027142 002312      BDATA
5875 027144 123737 002310 002312      CMPB     GDATA,BDATA      ;AND CHECK IT
5876 027152 001407      BEQ      10$      ;IF OK, PROCEED
5877 027154 012737 000007 002334      MOV      #7,REGNUM      ;IDENTIFY THE FAILING REGISTER &
5878 027162      GEDF     EM20,ERR7      ; REPORT FAILURE
5879      ;          'DEVICE FATAL' ERROR # 40
5880 027162 104455      TRAP      C$ERDF
5881 027164 000050      .WORD    40
5882 027166 015243      .WORD    EM20
5883 027170 006612      .WORD    ERR7
5884
5885      ; ***** STEP D *****
5886
5887 027172 004537 004064      10$: JSR      R5,READ      ;READ T1L-L(ADDR 06)
5888 027176 120006      T1LL
5889 027200 002312      BDATA
5890 027202 105737 002312      TSTB     BDATA      ;AND CHECK IT
5891 027206 001411      BEQ      2$      ;IF OK, PROCEED
5892 027210 005037 002310      CLR      GDATA
5893 027214 012737 000006 002334      MOV      #6,REGNUM      ;IDENTIFY THE FAILING REGISTER &
5894 027222      GEDF     EM20B,ERR7      ; REPORT FAILURE
5895      ;          'DEVICE FATAL' ERROR # 41
5896 027222 104455      TRAP      C$ERDF
5897 027224 000051      .WORD    41
5898 027226 015355      .WORD    EM20B
5899 027230 006612      .WORD    ERR7
5900
5901      ; ***** STEP E *****
5902
5903 027232      2$:  ENDSEG
5904 027232
5905 027232 104405      10000$: TRAP      C$ESEG
5906
5907 027234 000402      BR      21$

```

CVDMAA.P11 12-DEC-80 15:59

TEST 15 -- VIA'S T1 DATA READ/WRITE

5908 027236 000137 027100  
 5909 027242 077303  
 5910  
 5911 027244  
 5912 027244  
 5913 027244 104403  
 5914  
 5915 027246  
 5916 027246  
 5917 027246 104401

20\$: JMP T16,L2  
 21\$: SOB R3,20\$

;IF MORE DATA, DO ANOTHER BYTE  
 ;ELSE, EXIT SUBTEST

ENDSUB

L10053: TRAP C\$ESUB

ENDTST

L10050: TRAP C\$ETST



CVDMAA.P11 12-DEC-80 15:59

TEST 16 -- VIA'S SR DATA READ/WRITE

.SBTTL TEST 16 -- VIA'S SR DATA READ/WRITE

```

:*****
:*
:* TEST 16 -- VIA'S SR DATA READ/WRITE
:*
:* SR == 'SHIFT REGISTER'
:*
:* FIRST, A MASTER CLEAR IS PERFORMED AND THE ACR IS SET TO 000. THEN :
:* READ/WRITE BITS 0-7 OF VIA SHIFT REGISTER ARE TESTED BY WRITING, READING,
:* AND COMPARING EACH BYTE OF DATA PATTERN B.
:* DATA PATTERN B = 125, 252, 000, 377, 001, 002, 004, 010, 020, 040, 100,
:*                   200, 376, 375, 373, 367, 357, 337, 277, 177, 000
:*****

```

```

5918
5919
5920
5921
5922
5923
5924
5925
5926
5927
5928
5929
5930
5931
5932
5933
5934
5935 027250
5936 027250 004737 003762
5937 027254 103003
5938 027256
5939 027256 104460
5940 027260
5941 027260 104410
5942 027262 000046
5943
5944 027264 012701 002526
5945 027270 012103
5946
5947 027272
5948 027272
5949 027272
5950 027272 104402
5951
5952 027274 111137 002306
5953 027300 112137 002310
5954 027304 012700 120012
5955 027310 004737 005034
5956 027314 103003
5957 027316
5958 027316 104460
5959 027320
5960 027320 104410
5961 027322 000006
5962
5963 027324
5964 027324
5965 027324 104403
5966
5967 027326 077317
5968
5969
5970 027330
5971 027330
5972 027330 104401

```

```

:
: BGNTST
:
: JSR PC,MSTCLR ;INIT DMV & START UP THE MAINT. LOOP
: BCC 30$ ;IF AN ERROR OCCURED,
: ERROR ;REPORT IT &
: ESCAPE TST ; EXIT TRAP C$ERROR
:
: 30$: MOV #PATB,R1 ;POINT TO PATTERN TABLE
: MOV (R1)+,R3 ;GET # OF ENTRIES IN TABLE
:
: T18.LP: BGNSUB ;THE SUBTEST ONLY TESTS THE ONE PATTERN
:
: T16.1: TRAP C$BSUB
:
: MOVB (R1),TDATA ;SETUP TEST DATA BYTE FOR 'STREG'
: MOVB (R1)+,GDATA ;SETUP EXPECTED DATA BYTE FOR 'STREG'
: MOV #SR,R0 ;SPECIFY THE REGISTER BEING TESTED
: JSR PC,STREG ;PERFORM STATIC TEST OF THE SPECIFIED REGISTER
: BCC 10$ ;WAS AN ERROR FOUND?
: ERROR ;YES, REPORT IT AND
: ESCAPE TST ; EXIT FROM THE TEST. 'CKLOOP' IS IMPLIED
: TRAP C$ERROR
: TRAP C$ESCAPE
: .WORD L10054-.
:
: 10$: ENDSUB
:
: SOB R3,T18.LP ;IF THERE IS IN FACT MORE DATA, LOOP BACK TO
: ;TEST IT. ELSE, FALL OUT OF LOOP AND TEST
:
: ENDTST
:
: L10055: TRAP C$ESUB
:
: L10054: TRAP C$ETST

```

CVDMAA.P11 12-DEC-80 15:59

TEST 17 -- VIA'S ACR DATA READ/WRITE

.SBTTL TEST 17 -- VIA'S ACR DATA READ/WRITE

```

*****
*
* TEST 17 -- VIA'S ACR DATA READ/WRITE
*
* ACR == 'AUXILIARY CONTROL REGISTER'
*
* FIRST, A MASTER CLEAR IS PERFORMED. THEN :
* READ/WRITE BITS 0-7 OF THE ACR ARE TESTED BY WRITING, READING,
* AND COMPARING EACH BYTE OF DATA PATTERN B.
* DATA PATTERN B = 125, 252, 000, 377, 001, 002, 004, 010, 020, 040, 100,
*                   200, 376, 375, 373, 367, 357, 337, 277, 177, 000
*
*****

```

```

5973
5974
5975
5976
5977
5978
5979
5980
5981
5982
5983
5984
5985
5986
5987
5988
5989
5990 027332
5991 027332 004737 003762
5992 027336 103003
5993 027340
5994 027340 104460
5995 027342
5996 027342 104410
5997 027344 000046
5998
5999 027346 012701 002526
6000 027352 012103
6001
6002 027354
6003 027354
6004 027354
6005 027354 104402
6006
6007 027356 111137 002306
6008 027362 112137 002310
6009 027366 012700 120013
6010 027372 004737 005034
6011 027376 103003
6012 027400
6013 027400 104460
6014 027402
6015 027402 104410
6016 027404 000006
6017
6018 027406
6019 027406
6020 027406 104403
6021
6022 027410 077317
6023
6024
6025 027412
6026 027412
6027 027412 104401

```

```

:
: BGNTST
:
: JSR PC,MSTCLR ;INIT DMV & START UP THE MAINT. LOOP
: BCC 30$ ;IF AN ERROR OCCURED,
: ERROR ;REPORT IT &
: ESCAPE TST ; EXIT TRAP C$ERROR
:
: 30$: MOV #PATB,R1 ;POINT TO PATTERN TABLE
: MOV (R1)+,R3 ;GET # OF ENTRIES IN TABLE
:
: T19.LP:
: BGNSUB ;THE SUBTEST ONLY TESTS THE ONE PATTERN
:
: MOVB (R1),TDATA ;SETUP TEST DATA BYTE FOR 'STREG'
: MOVB (R1)+,GDATA ;SETUP EXPECTED DATA BYTE FOR 'STREG'
: MOV #ACR,R0 ;SPECIFY THE REGISTER BEING TESTED
: JSR PC,STREG ;PERFORM STATIC TEST OF THE SPECIFIED REGISTER
: BCC 10$ ;WAS AN ERROR FOUND?
: ERROR ;YES, REPORT IT AND
: ESCAPE TST ; EXIT FROM THE TEST. 'CKLOOP' IS IMPLIED TRAP C$ERROR
:
: 10$: ENDSUB ;TRAP C$ESCAPE
:
: SOB R3,T19.LP ;IF THERE IS IN FACT MORE DATA, LOOP BACK TO
: ;TEST IT. ELSE, FALL OUT OF LOOP AND TEST
:
: ENDTST
:
: L10056: TRAP C$SETST
:
: L10057: TRAP C$ESUB
:
: L10056-.
:
: L10056-.

```



CVDMAA.P11 12-DEC-80 15:59

TEST 18 -- VIA'S PCR DATA READ/WRITE

.SBTTL TEST 18 -- VIA'S PCR DATA READ/WRITE

```

:*****
:*
:* TEST 18 -- VIA'S PCR DATA READ/WRITE
:*
:* PCR == 'PERIPHERAL CONTROL REGISTER'
:*
:* FIRST, A MASTER CLEAR IS PERFORMED. THEN :
:* READ/WRITE BITS 0-7 OF THE PCR REGISTER ARE TESTED BY WRITING, READING,
:* AND COMPARING EACH BYTE OF A SUBSET OF DATA PATTERN B.
:* DATA PATTERN B (SUBSET) = 125, 252, 000, 377, 001, 002, 004, 010, 020,
:*                               040, 100, 200.
:*****

```

6028  
6029  
6030  
6031  
6032  
6033  
6034  
6035  
6036  
6037  
6038  
6039  
6040  
6041  
6042  
6043  
6044  
6045  
6046  
6047  
6048  
6049  
6050  
6051  
6052  
6053  
6054  
6055  
6056  
6057  
6058  
6059  
6060  
6061  
6062  
6063  
6064  
6065  
6066  
6067  
6068  
6069  
6070  
6071  
6072  
6073  
6074  
6075  
6076  
6077  
6078  
6079  
6080  
6081

027414  
027414 004737 003762  
027420 103003  
027422  
027422 104460  
027424  
027424 104410  
027426 000050  
027430 012701 002530  
027434 012703 002543  
027440  
027440  
027440 104402  
027442 111137 002306  
027446 112137 002310  
027452 012700 120014  
027456 004737 005034  
027462 103003  
027464 104460  
027466  
027466 104410  
027470 000006  
027472  
027472  
027472 104403  
027474 077317  
027476  
027476  
027476 104401

```

:
: BGNTST
:
: JSR PC,MSTCLR ;INIT DMV & START UP THE MAINT. LOOP
: BCC 30$ ;IF AN ERROR OCCURED,
: ERROR ;REPORT IT &
:
: ESCAPE TST ; EXIT TRAP C$ERROR
:
: 30$: MOV #PATB+2,R1 ;POINT TO PATTERN TABLE
: MOV #PATB+15,R3 ;GET # OF ENTRIES IN TABLE
:
: T20.LP:
: BGNSUB ;THE SUBTEST ONLY TESTS THE ONE PATTERN
:
: MOVB (R1),TDATA ;SETUP TEST DATA BYTE FOR 'STREG'
: MOVB (R1)+,GDATA ;SETUP EXPECTED DATA BYTE FOR 'STREG'
: MOV #PCR,R0 ;SPECIFY THE REGISTER BEING TESTED
: JSR PC,STREG ;PERFORM STATIC TEST OF THE SPECIFIED REGISTER
: BCC 10$ ;WAS AN ERROR FOUND?
: ERROR ;YES, REPORT IT AND
:
: ESCAPE TST ; EXIT FROM THE TEST. 'CKLOOP' IS IMPLIED TRAP C$ERROR
:
: 10$: ENDSUB ; TRAP C$ESUB
:
: SOB R3,T20.LP ;IF THERE IS IN FACT MORE DATA, LOOP BACK TO
: ENDTST ;TEST IT. ELSE, FALL OUT OF LOOP AND TEST
:
: L10060: TRAP C$ETST
:
: L10061: TRAP C$ESUB
:
: T18::
: T18.1: TRAP C$BSUB
:
: .WORD L10060-.
:
: .WORD L10060-.

```

CVDMAA.P11 12-DEC-80 15:59

TEST 19 -- VIA'S IER DATA READ/WRITE

.SBTTL TEST 19 -- VIA'S IER DATA READ/WRITE

```

:*****
:*
:*          TEST 19 -- VIA'S IER DATA READ/WRITE
:*
:*          IER == 'INTERRUPT ENABLE REGISTER'
:*
:* BITS 0-6 OF THE IER CAN BE SET OR CLEARED ON A WRITE, UNDER CONTROL OF THE
:* SET/CLEAR CONTROL BIT 7. TO TEST THIS, EACH BYTE OF DATA PATTERN D IS
:* WRITTEN INTO IER, AND THE REGISTER IS READ AND COMPARED TO THE CORRESPOND-
:* ING BYTE OF DATA PATTERN E.
:*
:* DATA PATTERN D = 200, 201, 202, 204, 210, 220, 240, 300, 200, 000, 001,
:*                  002, 004, 010, 020, 040, 100, 000, 325, 125, 252, 052
:*
:* DATA PATTERN E = 000, 001, 003, 007, 017, 037, 077, 177, 177, 177, 176,
:*                  174, 170, 160, 140, 100, 000, 000, 125, 000, 052, 000

```

```

:*****
:
:          BGNTST
:
:                                     T19::
:          JSR      PC,MSTCLR           ;INIT DMV & START UP THE MAINT. LOOP
:          BCC      30$                 ;IF AN ERROR OCCURED,
:          ERROR                    ;REPORT IT &
:                                     TRAP      C$ERROR
:          ESCAPE   TST                  ; EXIT
:                                     TRAP      C$ESCAPE
:                                     .WORD     L10062-.
:
:30$:  MOV      #PATD,R1                ;POINT TO PATTERN TABLE
:       MOV      (R1)+,R3                ;GET # OF ENTRIES IN TABLE
:       MOV      #PATE+2,R2              ;POINT TO 'EXPECTED' DATA PATTERN TABLE
:
:T21.LP:
:       BGNSUB                          ;THE SUBTEST ONLY TESTS THE ONE PATTERN
:                                     T19.1:
:                                     TRAP      C$BSUB
:
:       MOVB     (R1)+,TDATA              ;SETUP TEST DATA BYTE FOR 'STREG'
:       MOVB     (R2)+,GDATA              ;SETUP EXPECTED DATA BYTE FOR 'STREG'
:       MOV      #IENR,R0                 ;SPECIFY THE REGISTER BEING TESTED
:       JSR      PC,STREG                 ;PERFORM STATIC TEST OF THE SPECIFIED REGISTER
:       BCC      10$                     ;WAS AN ERROR FOUND?
:       ERROR                    ;YES, REPORT IT AND
:                                     TRAP      C$ERROR
:          ESCAPE   TST                  ; EXIT FROM THE TEST. 'CKLOOP' IS IMPLIED
:                                     TRAP      C$ESCAPE
:                                     .WORD     L10062-.
:
:10$:  ENDSUB
:
:L10063:
:       TRAP      C$ESUB

```

```

6082
6083
6084
6085
6086
6087
6088
6089
6090
6091
6092
6093
6094
6095
6096
6097
6098
6099
6100
6101
6102
6103
6104
6105 027500
6106 027500 004737 003762
6107 027504 103003
6108 027506
6109 027506 104460
6110 027510
6111 027510 104410
6112 027512 000052
6113
6114 027514 012701 002644
6115 027520 012103
6116 027522 012702 002676
6117
6118 027526
6119 027526
6120 027526
6121 027526 104402
6122
6123 027530 112137 002306
6124 027534 112237 002310
6125 027540 012700 120016
6126 027544 004737 005034
6127 027550 103003
6128 027552
6129 027552 104460
6130 027554
6131 027554 104410
6132 027556 000006
6133
6134 027560
6135 027560
6136 027560 104403
6137

```



CVDMAA.P11 12-DEC-80 15:59

TEST 19 -- VIA'S IER DATA READ/WRITE

6138 027562 077317  
6139  
6140  
6141 027564  
6142 027564  
6143 027564 104401

SOB R3.T21.LP

;IF THERE IS IN FACT MORE DATA, LOOP BACK TO  
;TEST IT. ELSE, FALL OUT OF LOOP AND TEST

ENDTST

L10062: TRAP CSETST

CVDMAA.P11 12-DEC-80 15:59

TEST 20 -- VIA'S ORB/DDR8 MASTER CLEAR TEST

.SBTTL TEST 20 -- VIA'S ORB/DDR8 MASTER CLEAR TEST

6144  
6145  
6146  
6147  
6148  
6149  
6150  
6151  
6152  
6153  
6154  
6155  
6156  
6157  
6158  
6159  
6160  
6161  
6162  
6163  
6164  
6165  
6166  
6167  
6168  
6169  
6170  
6171  
6172  
6173  
6174  
6175  
6176  
6177  
6178  
6179  
6180  
6181  
6182  
6183  
6184  
6185  
6186  
6187  
6188  
6189  
6190  
6191  
6192  
6193  
6194  
6195  
6196  
6197  
6198  
6199

027566  
027566 004737 003762  
027572 103003  
027574 104460  
027576 104410  
027600 000252  
027602 012737 000377 002310  
027610 013737 002310 002306  
027616 012700 120002  
027622 004737 005034  
027626 103003  
027630 104460  
027632 104410  
027634 000216  
027636 012700 120000  
027642 004737 005034  
027646 103003  
027650 104460  
027652 104410  
027654 000176  
027656 004737 003762  
027662 103003  
027664

```

:*****
:
:   TEST 20 -- VIA'S ORB/DDR8 MASTER CLEAR TEST
:
:   ORB == 'OUTPUT REGISTER PORT B'
:   DDRB == 'DATA DIRECTION REGISTER B'
:
: FIRST, A MASTER CLEAR IS PERFORMED. NEXT, 377 IS LOADED INTO DDRB TO SET
: ALL B PORT PINS FOR OUTPUT MODE. THEN, A 000 BYTE IS WRITTEN INTO ORB AND
: THE REGISTER IS READ BACK AND CHECKED FOR 000. THEN, A MASTER CLEAR IS
: PERFORMED AND ORB IS READ AND CHECKED FOR 377.
:*****
:
:   BGNTST
:
:   T20::
:
:   JSR    PC,MSTCLR    ;INIT DMV & START UP THE MAINT. LOOP
:   BCC   1$           ;IF AN ERROR OCCURED,
:   ERROR                ;REPORT IT &
:
:   ESCAPE TST          ; EXIT
:
:   TRAP   C$ERROR
:
:   TRAP   C$ESCAPE
:   .WORD L10064-.
:
:   1$:   MOV    #377,GDATA    ;SETUP FOR CALL TO STREG
:   MOV    GDATA,TDATA
:
: WE'LL USE 'STREG' TO LOAD & CHECK 'DDR8' WITH 377 THEREBY SETTING UP
: 'ORB' FOR BI-DIRECTIONAL TRANSFERS
:
:   MOV    #DDR8,R0      ;POINT TO ORB
:   JSR    PC,STREG     ;LOAD & TEST IT
:   BCC   4$           ;IF OK, PROCEED WITH TESTING
:   ERROR                ;ELSE, REPORT THE ERROR
:
:   ESCAPE TST          ; & QUIT
:
:   TRAP   C$ERROR
:
:   TRAP   C$ESCAPE
:   .WORD L10064-.
:
: NOW WE'LL USE 'STREG' TO SET & CHECK 'ORB'
:
:   4$:   MOV    #ORB,R0    ;POINT TO DDRB
:   JSR    PC,STREG     ;LOAD & TEST 'ORB'
:   BCC   5$           ;IF NO ERROR HERE, PROCEED
:   ERROR                ;ELSE, REPORT THE ERROR
:
:   ESCAPE TST          ; & QUIT
:
:   TRAP   C$ERROR
:
:   TRAP   C$ESCAPE
:   .WORD L10064-.
:
:   5$:   JSR    PC,MSTCLR    ;ISSUE THE MASTER CLEAR (STAY IN M-LOOP)
:   BCC   10$          ;IF NO ERROR HERE, PROCEED
:   ERROR                ;ELSE, REPORT IT

```



CVDMAA.P11 12-DEC-80 15:59

TEST 20 -- VIA'S ORB/DDR8 MASTER CLEAR TEST

```

6200 027664 104460                                TRAP  C$ERROR
6201 027666                                ESCAPE TST ; & QUIT                                TRAP  C$ESCAPE
6202 027666 104410                                .WORD  L10064-.
6203 027670 000162
6204
6205 027672 005037 002310 10$: CLR  GDATA ;FOR TESTING PURPOSES LATER
6206
6207 027676 004537 004064 JSR  R5,READ ;READ THE 'RESET' VALUE OF THE 'DDR8'
6208 027702 120002 DDR8
6209 027704 002312 BDATA
6210 027706 103003 BCC 12$ ;IF NO ERROR READING IT, PROCEED
6211 027710 ERROR ;ELSE, REPORT IT
6212 027710 104460                                TRAP  C$ERROR
6213 027712                                ESCAPE TST ; & QUIT                                TRAP  C$ESCAPE
6214 027712 104410                                .WORD  L10064-.
6215 027714 000136
6216
6217 027716 123737 002312 002310 12$: CMPB BDATA,GDATA ;DID IT GET CLEARE?
6218 027724 001407 BEQ 14$ ;YES, GOOD. NOW CHECK 'ORB'
6219 027726 012737 000002 002334 MOV #DDR8<17>,REGNUM ;NO! BUILD REGISTER # POINTER
6220 027734 GEDF EM5,ERR7 ;REPORT MASTER CLEAR FAILURE
6221 ; 'DEVICE FATAL' ERROR # 42
6222 027734 104455                                TRAP  C$ERDF
6223 027736 000052                                .WORD  42
6224 027740 014515                                .WORD  EM5
6225 027742 006612                                .WORD  ERR7
6226
6227 027744 012737 000377 002310 14$: MOV #377,GDATA ;SETUP FOR CALL TO STREG
6228 027752 013737 002310 002306 MOV  GDATA,TDATA
6229
6230 ; WE'LL USE 'STREG' TO LOAD & CHECK 'DDR8' WITH 377 THEREBY SETTING UP
6231 ; 'ORB' FOR BY-DIRECTIONAL TRANSFERS
6232
6233 027760 012700 120002 MOV #DDR8,R0 ;POINT TO ORB
6234 027764 004737 005034 JSR  PC,STREG ;LOAD & TEST IT
6235 027770 103003 BCC 16$ ;IF OK, PROCEED WITH TESTING
6236 027772 ERROR ;ELSE, REPORT THE ERROR
6237 027772 104460                                TRAP  C$ERROR
6238 027774                                ESCAPE TST ; & QUIT                                TRAP  C$ESCAPE
6239 027774 104410                                .WORD  L10064-.
6240 027776 000054
6241
6242 030000 005037 002310 16$: CLR  GDATA ;SETUP FOR TESTING ORB
6243 030004 004537 004064 JSR  R5,READ ;NOW READ THE 'RESET' VALUE OF 'ORB'
6244 030010 120000 ORB
6245 030012 002312 BDATA
6246 030014 103003 BCC 18$ ;IF NO ERROR READING IT, PROCEED
6247 030016 ERROR ;ELSE, REPORT IT
6248 030016 104460                                TRAP  C$ERROR
6249 030020                                ESCAPE TST ; & QUIT                                TRAP  C$ESCAPE
6250 030020 104410                                .WORD  L10064-.
6251 030022 000030
6252
6253 030024 123737 002310 002312 18$: CMPB GDATA,BDATA ;WAS IT PROPERLY RESET?
6254 030032 001407 BEQ 32$ ;YES, THIS TEST IS DONE, EXIT
6255 030034 012737 000000 002334 MOV #ORB<17>,REGNUM ;NO! BUILD REGISTER # POINTER

```

CVDMAA.P11 12-DEC-80 15:59

TEST 20 VIA'S ORB/DDRB MASTER CLEAR TEST

6256	030042	
6257		
6258	030042	104455
6259	030044	000053
6260	030046	014515
6261	030050	006612
6262		
6263	030052	
6264	030052	
6265	030052	104401

GEDF EM5,ERR7

;REPORT MASTER CLEAR FAILURE  
; 'DEVICE FATAL' ERROR # 43

TRAP	C\$ERDF
.WORD	43
.WORD	EM5
.WORD	ERR7

32\$: ENDTST

L10064:

TRAP	C\$ETST
------	---------



CVDMAA.P11 12-DEC-80 15:59

TEST 21 -- VIA'S DDRB MASTER CLEAR TEST

.SBTTL TEST 21 -- VIA'S DDRB MASTER CLEAR TEST

6266  
6267  
6268  
6269  
6270  
6271  
6272  
6273  
6274  
6275  
6276  
6277  
6278  
6279  
6280  
6281  
6282  
6283  
6284  
6285  
6286  
6287  
6288  
6289  
6290  
6291  
6292  
6293  
6294  
6295  
6296  
6297  
6298  
6299  
6300  
6301  
6302  
6303  
6304  
6305  
6306  
6307  
6308  
6309  
6310  
6311  
6312  
6313  
6314  
6315  
6316  
6317  
6318  
6319  
6320  
6321

030054

030054 004737 003762  
030060 103003  
030062 104460  
030064 104410  
030066 000114  
030070 012737 000377 002310  
030076 013737 002310 002306  
030104 012700 120002  
030110 004737 005034  
030114 103003  
030116 104460  
030120 104410  
030122 000060  
030124 004737 003762  
030130 103003  
030132 104460  
030134 104410  
030136 000044  
030140 005037 002310  
030144 004537 004064  
030150 120002  
030152 002312  
030154 123737 002310 002312

```
*****
*
* TEST 21 -- VIA'S DDRB MASTER CLEAR TEST
*
* DDRB == 'DATA DIRECTION REGISTER B'
*
* A 377 BYTE IS WRITTEN INTO DDRB AND THE REGISTER IS READ BACK AND CHECKED
* FOR 377. THEN, A MASTER CLEAR IS PERFORMED AND DDRB IS READ AND CHECKED FOR
* 000.
*
* NOTE: THIS TESTING IS ALSO DONE IN TEST 23. IT IS INCLUDED HERE ONLY TO
* PROVIDE TIGHTER LOOPING ON JUST THE DDRB MASTER CLEAR CHECKING.
*****
```

```

: BGN:ST
:
: T21::
:
: JSR PC,MSTCLR ;INIT DMV & START UP THE MAINT. LOOP
: BCC 1$ ;IF AN ERROR OCCURED,
: ERROR ;REPORT IT &
:
: ESCAPE TST ; EXIT TRAP C$ERROR
:
: ESCAPE TST ; EXIT TRAP C$ESCAPE
: .WORD L10065-.
:
1$: MOV #377,GDATA ;SETUP FOR CALL TO STREG
: MOV GDATA,TDATA
: MOV #DDR,B,R0
:
: NOW WE'LL USE 'STREG' TO SET & CHECK 'DDR'B'
:
: JSR PC,STREG ;LOAD & TEST 'DDR'B'
: BCC 5$ ;IF NO ERROR HERE, PROCEED
: ERROR ;ELSE, REPORT IT
:
: ESCAPE TST ; & QUIT TRAP C$ERROR
:
: ESCAPE TST ; & QUIT TRAP C$ESCAPE
: .WORD L10065-.
:
5$: JSR PC,MSTCLR ;ISSUE THE MASTER CLEAR (STAY IN M-LOOP)
: BCC 10$ ;IF NO ERROR HERE, PROCEED
: ERROR ;ELSE, REPORT IT
:
: ESCAPE TST ; & QUIT TRAP C$ERROR
:
: ESCAPE TST ; & QUIT TRAP C$ESCAPE
: .WORD L10065-.
:
10$: CLR GDATA ;FOR TESTING PURPOSES LATER
: JSR R5,READ ;NOW READ THE 'RESET' VALUE OF 'DDR'B'
: DDRB
: BDATA
:
: CMPB GDATA,BDATA ;WAS IT PROPERLY RESET?
```

CVDMAA.P11 12-DEC-80 15:59

TEST 21 -- VIA'S DDRB MASTER CLEAR TEST

6322 030162 001407  
 6323 030164 012737 000002 002334  
 6324 030172  
 6325  
 6326 030172 104455  
 6327 030174 000054  
 6328 030176 014515  
 6329 030200 006612  
 6330  
 6331 030202  
 6332 030202  
 6333 030202 104401

BEQ 32\$ ;YES, THIS TEST IS DONE, EXIT  
 MOV #DDRBB<17>,REGNUM ;NO! BUILD REGISTER # POINTER  
 GEDF EMS,ERR7 ;REPORT MASTER CLEAR FAILURE  
 ; 'DEVICE FATAL' ERROR # 44

TRAP C\$ERDF  
 .WORD 44  
 .WORD EMS  
 .WORD ERR7

32\$: ENDTST

L10065:  
 TRAP C\$ETST





CVDMAA.P11 12-DEC-80 15:59

TEST 22 -- VIA'S DDRA MASTER CLEAR TEST

6390		
6391	030322	104455
6392	030324	000055
6393	030326	014515
6394	030330	006612
6395		
6396	030332	
6397	030332	
6398	030332	104401

; 'DEVICE FATAL' ERROR # 45

TRAP	C\$ERDF
.WORD	45
.WORD	EM5
.WORD	ERR7

32\$: ENDTST

L10066:

TRAP	C\$ETST
------	---------



CVDMAA.P11 12-DEC-80 15:59

TEST 23 -- VIA'S SR MASTER CLEAR TEST

.SBTTL TEST 23 -- VIA'S SR MASTER CLEAR TEST

```

+*****
*
* TEST 23 -- VIA'S SR MASTER CLEAR TEST
*
* SR == 'SHIFT REGISTER'
*
* A 123 BYTE IS WRITTEN INTO SR AND THE REGISTER IS READ BACK AND CHECKED
* FOR 123. THEN, A MASTER CLEAR IS PERFORMED AND SR IS READ AND CHECKED FOR
* NO CHANGE.
*
-----*****

```

```

6399
6400
6401
6402
6403
6404
6405
6406
6407
6408
6409
6410
6411
6412
6413
6414 030334
6415
6416 030334 004737 003762
6417 030340 103003
6418 030342
6419 030342 104460
6420 030344
6421 030344 104410
6422 030346 000120
6423
6424 030350 004537 004322
6425 030354 120013
6426 030356 000000
6427 030360 012737 000123 002310
6428 030366 013737 002310 002306
6429 030374 012700 120012
6430
6431
6432
6433 030400 004737 005034
6434 030404 103003
6435 030406
6436 030406 104460
6437 030410
6438 030410 104410
6439 030412 000054
6440
6441 030414 004737 003762
6442 030420 103003
6443 030422
6444 030422 104460
6445 030424
6446 030424 104410
6447 030426 000040
6448
6449 030430 004537 004064
6450 030434 120012
6451 030436 002312
6452
6453 030440 123737 002310 002312
6454 030446 001407

```

```

: BGNTST
:
: T23::
:
: JSR PC,MSTCLR ;INIT DMV & START UP THE MAINT. LOOP
: BCC 1$ ;IF AN ERROR OCCURED,
: ERROR ;REPORT IT &
:
: ESCAPE TST ; EXIT TRAP C$ERROR
:
: JSR R5,WRITEI ;FORCE SR TO MODE 0
: ACR
: 0
: MOV #123,GDATA ;SETUP FOR CALL TO STREG
: MOV GDATA,TDATA
: MOV #SR,R0
:
: NOW WE'LL USE 'STREG' TO SET & CHECK 'SR'
:
: JSR PC,STREG ;LOAD & TEST 'SR'
: BCC 5$ ;IF NO ERROR HERE, PROCEED
: ERROR ;ELSE, REPORT IT
:
: ESCAPE TST ; & QUIT TRAP C$ERROR
:
: JSR PC,MSTCLR ;ISSUE THE MASTER CLEAR (STAY IN M-LOOP)
: BCC 10$ ;IF NO ERROR HERE, PROCEED
: ERROR ;ELSE, REPORT IT
:
: ESCAPE TST ; & QUIT TRAP C$ERROR
:
: JSR R5,READ ;NOW READ THE 'RESET' VALUE OF 'SR'
: SR ; (IT SHOULDN'T HAVE CHANGED)
: BDATA
:
: CMPB GDATA,BDATA ;WAS IT PROPERLY RESET?
: BEQ 32$ ;YES, THIS TEST IS DONE, EXIT

```

CVDMAA.P11 12-DEC-80 15:59

TEST 23 -- VIA'S SR MASTER CLEAR TEST

6455 030450 012737 000012 002334  
 6456 030456  
 6457  
 6458 030456 104455  
 6459 030460 000056  
 6460 030462 014515  
 6461 030464 006612  
 6462  
 6463 030466  
 6464 030466  
 6465 030466 104401

MOV #SR8<17>,REGNUM ;NO!  
 GEDF EMS,ERR7 ;REPORT MASTER CLEAR FAILURE  
 ; 'DEVICE FATAL' ERROR # 46

TRAP CSEDF  
 .WORD 46  
 .WORD EMS  
 .WORD ERR7

32\$: ENDTST

L10067:

TRAP CSETST



CVDMAA.P11 12-DEC-80 15:59

TEST 24 -- VIA'S ACR MASTER CLEAR TEST

.SBTTL TEST 24 -- VIA'S ACR MASTER CLEAR TEST

```

:*****
:*
:* TEST 24 -- VIA'S ACR MASTER CLEAR TEST
:* ACR == 'AUXILIARY CONTROL REGISTER'
:* A 252 BYTE IS WRITTEN INTO ACR AND THE REGISTER IS READ BACK AND CHECKED
:* FOR 252. THEN, A MASTER CLEAR IS PERFORMED AND ACR IS READ AND CHECKED FOR
:* 000, TO VERIFY THAT IT IS CLEARED BY MASTER CLEAR.
:*****

```

6466  
6467  
6468  
6469  
6470  
6471  
6472  
6473  
6474  
6475  
6476  
6477  
6478  
6479  
6480  
6481  
6482  
6483  
6484  
6485  
6486  
6487  
6488  
6489  
6490  
6491  
6492  
6493  
6494  
6495  
6496  
6497  
6498  
6499  
6500  
6501  
6502  
6503  
6504  
6505  
6506  
6507  
6508  
6509  
6510  
6511  
6512  
6513  
6514  
6515  
6516  
6517  
6518  
6519  
6520  
6521

030470

030470 004737 003762  
030474 103003  
030476 104460  
030476 104460  
030500 104410  
030502 000114

030504 012737 000252 002310  
030512 013737 002310 002306  
030520 012700 120013

030524 004737 005034  
030530 103003  
030532 104460  
030534 104410  
030536 000060

030540 004737 003762  
030544 103003  
030546 104460  
030550 104410  
030552 000044

030554 005037 002310  
030560 004537 004064  
030564 120013  
030566 002312

030570 123.37 002310 002312  
030576 001407  
030600 012737 000013 002334  
030606

BGNTST

T24::

```

JSR PC,MSTCLR ;INIT DMV & START UP THE MAINT. LOOP
BCC 1$ ;IF AN ERROR OCCURED,
ERROR ;REPORT IT &
ESCAPE TST ; EXIT TRAP C$ERROR
; .WORD C$ESCAPE
; L10070-.

1$: MOV #252,GDATA ;SETUP FOR CALL TO STREG
MOV GDATA,TDATA
MOV #ACR,R0

; NOW WE'LL USE 'STREG' TO SET & CHECK 'ACR'

JSR PC,STREG ;LOAD & TEST 'ACR'
BCC 5$ ;IF NO ERROR HERE, PROCEED
ERROR ;ELSE, REPORT IT
ESCAPE TST ; & QUIT TRAP C$ERROR
; .WORD C$ESCAPE
; L10070-.

5$: JSR PC,MSTCLR ;ISSUE THE MASTER CLEAR (STAY IN M-LOOP)
BCC 10$ ;IF NO ERROR HERE, PROCEED
ERROR ;ELSE, REPORT IT
ESCAPE TST ; & QUIT TRAP C$ERROR
; .WORD C$ESCAPE
; L10070-.

10$: CLR GDATA ;FOR TESTING PURPOSES LATER
JSR R5,READ ;NOW READ THE 'RESET' VALUE OF 'ACR'
ACR
BDATA

CMPB GDATA,BDATA ;WAS IT PROPERLY RESET?
BEQ 32$ ;YES, THIS TEST IS DONE, EXIT
MOV #ACR<17>,REGNUM ;NO! BUILD REGISTER # POINTER
GEDF EM5,ERR7 ;REPORT MASTER CLEAR FAILURE

```

CVDMAA.P11 12-DEC-80 15:59

TEST 24 -- VIA'S ACR MASTER CLEAR TEST

6522		
6523	030606	104455
6524	030610	000057
6525	030612	014515
6526	030614	006612
6527		
6528	030616	
6529	030616	
6530	030616	104401

32\$: ENDTST

```

; 'DEVICE FATAL' ERROR # 47
TRAP C$ERDF
.WORD 47
.WORD EM5
.WORD ERR7

```

```

L10070:
TRAP C$ETST

```



CVDMAA.P11 12-DEC-80 15:59

TEST 25 -- VIA'S PCR MASTER CLEAR TEST

.SBTTL TEST 25 -- VIA'S PCR MASTER CLEAR TEST

6531  
6532  
6533  
6534  
6535  
6536  
6537  
6538  
6539  
6540  
6541  
6542  
6543  
6544  
6545  
6546  
6547  
6548  
6549  
6550  
6551  
6552  
6553  
6554  
6555  
6556  
6557  
6558  
6559  
6560  
6561  
6562  
6563  
6564  
6565  
6566  
6567  
6568  
6569  
6570  
6571  
6572  
6573  
6574  
6575  
6576  
6577  
6578  
6579  
6580  
6581  
6582  
6583  
6584  
6585  
6586

030620

030620 004737 003762  
030624 103003  
030626 104460  
030630 104410  
030632 000114  
030634 012737 000377 002310  
030642 013737 002310 002306  
030650 012700 120014  
030654 004737 005034  
030660 103003  
030662 104460  
030664 104410  
030666 000060  
030670 004737 003762  
030674 103003  
030676 104460  
030700 104410  
030702 000044  
030704 005037 002310  
030710 004537 004064  
030714 120014  
030716 002312  
030720 123737 002310 002312  
030726 001407  
030730 012737 000014 002334  
030736

```
*****
*
* TEST 25 -- VIA'S PCR MASTER CLEAR TEST
*
* PCR == 'PERIPHERAL CONTROL REGISTER'
*
* A 377 BYTE IS WRITTEN INTO PCR AND THE REGISTER IS READ BACK AND CHECKED
* FOR 377. THEN, A MASTER CLEAR IS PERFORMED AND PCR IS READ AND CHECKED FOR
* 000.
*****
```

```

: BGNTST
:
: 25::
: JSR PC,MSTCLR ;INIT DMV & START UP THE MAINT. LOOP
: BCC 1$ ;IF AN ERROR OCCURED,
: ERROR ;REPORT IT &
: ESCAPE TST ; EXIT TRAP CSERROR
: .WORD CS$ESCAPE
: L10071-.
1$: MOV #377,GDATA ;SETUP FOR CALL TO STREG
: MOV GDATA,TDATA
: MOV #PCR,R0
:
: NOW WE'LL USE 'STREG' TO SET & CHECK 'PCR'
: JSR PC,STREG ;LOAD & TEST 'PCR'
: BCC 5$ ;IF NO ERROR HERE, PROCEED
: ERROR ;ELSE, REPORT IT
: ESCAPE TST ; & QUIT TRAP CSERROR
: .WORD CS$ESCAPE
: L10071-.
5$: JSR PC,MSTCLR ;ISSUE THE MASTER CLEAR (STAY IN M-LOOP)
: BCC 10$ ;IF NO ERROR HERE, PROCEED
: ERROR ;ELSE, REPORT IT
: ESCAPE TST ; & QUIT TRAP CSERROR
: .WORD CS$ESCAPE
: L10071-.
10$: CLR GDATA ;FOR TESTING PURPOSES LATER
: JSR R5,READ ;NOW READ THE 'RESET' VALUE OF 'PCR'
: PCR
: BDATA
: CMPB GDATA,BDATA ;WAS IT PROPERLY RESET?
: BEQ 32$ ;YES, THIS TEST IS DONE, EXIT
: MOV #PCR<17>,REGNUM ;NO! BUILD REGISTER # POINTER
: GEDF EMS,ERR7 ;REPORT MASTER CLEAR FAILURE
```

CVDMAA.P11 12-DEC-80 15:59

TEST 25 -- VIA'S PCR MASTER CLEAR TEST

6587		
6588	030736	104455
6589	030740	000060
6590	030742	014515
6591	030744	006612
6592		
6593	030746	
6594	030746	
6595	030746	104401

```

; 'DEVICE FATAL' ERROR # 48
TRAP C$ERDF
.WORD 48
.WORD EMS
.WORD ERR7

```

32\$: ENDTST

```

L10071:
TRAP C$ETST

```



CVDMAA.P11 12-DEC-80 15:59

TEST 26 -- VIA'S IER MASTER CLEAR TEST

.SBTTL TEST 26 -- VIA'S IER MASTER CLEAR TEST

6596  
6597  
6598  
6599  
6600  
6601  
6602  
6603  
6604  
6605  
6606  
6607  
6608  
6609  
6610  
6611  
6612  
6613  
6614  
6615  
6616  
6617  
6618  
6619  
6620  
6621  
6622  
6623  
6624  
6625  
6626  
6627  
6628  
6629  
6630  
6631  
6632  
6633  
6634  
6635  
6636  
6637  
6638  
6639  
6640  
6641  
6642  
6643  
6644  
6645  
6646  
6647  
6648  
6649  
6650  
6651

030750

030750 004737 003762

030754 103003

030756

030756 104460

030760

030760 104410

030762 000122

030764 105077 151362

030770 012737 000377 002310

030776 013737 002310 002306

031004 012700 120016

031010 004737 005034

031014 103003

031016

031016 104460

031020

031020 104410

031022 000062

031024 004737 003762

031030 103003

031032

031032 104460

031034

031034 104410

031036 000046

031040 012737 000200 002310

031046 004537 004064

031052 120016

031054 002312

031056 123737 002310 002312

031064 001407

\*\*\*\*\*  
\*  
\* TEST 26 -- VIA'S IER MASTER CLEAR TEST  
\*  
\* IER == 'INTERRUPT ENABLE REGISTER'  
\*  
\* A 377 BYTE IS WRITTEN INTO IER AND THE REGISTER IS READ BACK AND CHECKED  
\* FOR 377. THEN, A MASTER CLEAR IS PERFORMED AND IER IS READ AND CHECKED FOR  
\* 200.  
\*  
\*\*\*\*\*

```

:      BGNTST
:
:      T26::
:
:      JSR    PC,MSTCLR      ;INIT DMV & START UP THE MAINT. LOOP
:      BCC   1$             ;IF AN ERROR OCCURED,
:      ERROR 1$             ;REPORT IT &
:
:      ESCAPE TST           ; EXIT
:
:      TRAP   CSERROR
:      .WORD L10072-.
:
:      1$:   CLRB    @BSELO  ;MAKE SURE NO Q-BUS INTERRUPTS RESULT FROM
:
:      MOV    #377,GDATA    ; TESTING THE IER REGISTER
:      MOV    GDATA,TDATA  ;SETUP FOR CALL TO STREG
:      MOV    #IENR,R0
:
:      ; NOW WE'LL USE 'STREG' TO SET & CHECK 'IER'
:
:      JSR    PC,STREG      ;LOAD & TEST 'IER'
:      BCC   5$             ;IF NO ERROR HERE, PROCEED
:      ERROR 5$             ;ELSE, REPORT IT
:
:      ESCAPE TST           ; & QUIT
:
:      TRAP   CSERROR
:      .WORD L10072-.
:
:      5$:   JSR    PC,MSTCLR ;ISSUE THE MASTER CLEAR (STAY IN M-LOOP)
:      BCC   10$            ;IF NO ERROR HERE, PROCEED
:      ERROR 10$           ;ELSE, REPORT IT
:
:      ESCAPE TST           ; & QUIT
:
:      TRAP   CSERROR
:      .WORD L10072-.
:
:      10$:  MOV    #200,GDATA ;FOR TESTING PURPOSES LATER
:      JSR    R5,READ      ;NOW READ THE 'RESET' VALUE OF 'IER'
:      IENR  BDATA
:
:      CMPB  GDATA,BDATA   ;WAS IT PROPERLY RESET?
:      BEQ   32$           ;YES, THIS TEST IS DONE, EXIT

```

CVDMAA.P11 12-DEC-80 15:59

TEST 26 -- VIA'S IER MASTER CLEAR TEST

6652 031066 012737 000016 002334  
 6653 031074  
 6654  
 6655 031074 104455  
 6656 031076 000061  
 6657 031100 014515  
 6658 031102 006612  
 6659  
 6660 031104  
 6661 031104  
 6662 031104 104401  
 6663

MOV #IENR8<17>,REGNUM ;NO! BUILD REGISTER # POINTER  
 GEDF EMS,ERR7 ;REPORT MASTER CLEAR FAILURE  
 ; 'DEVICE FATAL' ERROR # 49

TRAP C\$ERDF  
 .WORD 49  
 .WORD EMS  
 .WORD ERR7

32\$: ENDTST

L10072:

TRAP C\$ETST



CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

.SBTTL TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

6664  
6665  
6666  
6667  
6668  
6669  
6670  
6671  
6672  
6673  
6674  
6675  
6676  
6677  
6678  
6679  
6680  
6681  
6682  
6683  
6684  
6685  
6686  
6687  
6688  
6689  
6690  
6691  
6692  
6693  
6694  
6695  
6696  
6697  
6698  
6699  
6700  
6701  
6702  
6703  
6704  
6705  
6706  
6707  
6708  
6709  
6710  
6711  
6712  
6713  
6714  
6715  
6716  
6717  
6718  
6719

031106  
031106  
031106  
031106 104402

```

*****
*
*   TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE
*
* THIS TEST VERIFIES THAT THE TIMER 1 COUNTER IS OPERATIONAL IN ONE-SHOT
* MODE, IN EACH OF TWO SUBTESTS.
*
* IN THE FIRST SUBTEST, THE FOLLOWING IS PERFORMED :
*
*   A MASTER CLEAR IS DONE AND THE TIMER IS PLACED IN ONE-SHOT MODE BY
*   SETTING BOTH ACR7 & ACR6 TO 0.
*
*   THE PROGRAM CHECKS FOR THE 'T1' FLAG (BIT 6) IN THE IFR TO BE
*   INITIALLY CLEARED.
*
*   T1L-L (ADR 04) & T1C-H (ADR 05) ARE BOTH LOADED WITH 252 (OCTAL).
*   (THIS IS EQUIVALENT TO AAAA (HEX) OR 43,690 (DECIMAL).) LOADING
*   T1C-H STARTS THE COUNTER.
*
*   THE PROGRAM PERIODICALLY CHECKS THE COUNTER TO VERIFY THAT IT IS
*   DECREMENTING AND THAT IT EVENTUALLY UNDERFLOWS PAST 0 AND CONTINUES
*   TO DECREMENT.
*
*   T1L-L (ADR 04) IS LOADED WITH 001 & T1C-H (ADR 05) IS LOADED WITH
*   000 IN ORDER TO SET 'T1' WITH A QUICK UNDERFLOW. THE 'T1' FLAG BIT
*   IN IFR IS READ AND CHECKED TO BE SET.
*
*   T1C-H, T1L-L, & T1L-H (ADDR'S 05, 06, & 07 RESP.) ARE READ AND AFTER
*   EACH THE 'T1' INTERRUPT FLAG IS CHECKED TO BE STILL SET.
*
*   T1C-L (ADDR 04) IS READ AND 'T1' IS CHECKED TO BE CLEARED.
*
*   T1C-H IS LOADED WITH 0 AGAIN TO INITIATE A NEW COUNT DOWN (WHICH
*   SHOULD UNDERFLOW ALMOST IMMEDIATELY) AND THE 'T1' BIT IN IFR IS
*   CHECKED TO BE SET AGAIN.
*
*   T1L-L IS LOADED WITH 125 (OCTAL) AND 'T1' BIT IS CHECKED TO BE STILL
*   SET.
*
*   T1C-H IS LOADED WITH 125, AND THE 'T1' BIT IS READ AND CHECKED TO BE
*   CLEARED BY THE LOADING OF T1C-H.
*
* IN THE SECOND SUBTEST, ALL OF THE ABOVE OPERATIONS ARE REPEATED,
* WITH ACR BIT 7 SET TO 1, AND ACR BIT 6 SET TO 0. 'PB7' (BIT 7 OF ORB)
* WILL BE MONITORED FOR ITS EXPECTED LEVELS DURING THIS SUBTEST.

```

BGNTST  
BGNSUB

T27::  
T27.1: TRAP CSBSUB

CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

6720
6721
6722 031110 004737 003762      JSR    PC,MSTCLR      ;INIT DMV & ENTER M-LOOP
6723 031114 103003              BCC    1$            ;IF NO ERROR, PROCEED WITH TESTING
6724 031116                      ERROR                   ;ELSE, REPORT ERROR
6725 031116 104460              ESCAPE TST           ;   & EXIT TEST
6726 031120                      TRAP    C$ERROR
6727 031120 104410              TRAP    C$ESCAPE
6728 031122 004734              .WORD  L10073-.
6729 031124 004537 004660      1$:  JSR    R5,INITT1   ;INITIALIZE TIMER # 1
6730 031130 000000              ;   0 ==> LATCHES
6731 031132 000000              ;   MODE 0 & 'T1' INT. ENABLE FLAG CLEARED
6732 031134 103003              BCC    .+10         ;IF NO ERROR, PROCEED
6733 031136                      ERROR                   ;ELSE, REPORT IT
6734 031136 104460              ESCAPE TST           ;   AND EXIT THIS TEST
6735 031140                      TRAP    C$ERROR
6736 031140 104410              TRAP    C$ESCAPE
6737 031142 004714              .WORD  L10073-.
6738 031144 004737 036112      JSR    PC,GETT1      ;IS 'T1' SET?
6739 031150 102002              BVC    .+6          ;IF NO ERROR, PROCEED
6740 031152                      ESCAPE SUB           ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
6741 031152 104410              TRAP    C$ESCAPE
6742 031154 002202              .WORD  L10074-.
6743 031156 103143              BCC    6$           ;NO, GOOD.
6744 031160 004537 004064      JSR    R5,READ      ;GET T1CL FOR ERROR MESSAGE
6745 031164 120004              T1CL
6746 031166 002450              TMP4
6747 031170 103003              BCC    .+10         ;IF NO ERROR, PROCEED
6748 031172                      ERROR                   ;ELSE, REPORT IT
6749 031172 104460              ESCAPE TST           ;   AND EXIT THIS TEST
6750 031174                      TRAP    C$ERROR
6751 031174 104410              TRAP    C$ESCAPE
6752 031176 004660              .WORD  L10073-.
6753 031200 004537 004064      JSR    R5,READ      ;GET T1CH FOR ERROR MESSAGE
6754 031204 120005              T1CH
6755 031206 002452              TMP5
6756 031210 103003              BCC    .+10         ;IF NO ERROR, PROCEED
6757 031212                      ERROR                   ;ELSE, REPORT IT
6758 031212 104460              ESCAPE TST           ;   AND EXIT THIS TEST
6759 031214                      TRAP    C$ERROR
6760 031214 104410              TRAP    C$ESCAPE
6761 031216 004640              .WORD  L10073-.
6762 031220 004537 004064      JSR    R5,READ      ;GET T1LL FOR ERROR MESSAGE
6763 031224 120006              T1LL
6764 031226 002454              TMP6
6765 031230 103003              BCC    .+10         ;IF NO ERROR, PROCEED
6766 031232                      ERROR                   ;ELSE, REPORT IT
6767 031232 104460              ESCAPE TST           ;   AND EXIT THIS TEST
6768 031234                      TRAP    C$ERROR
6769 031234 104410              TRAP    C$ESCAPE
6770 031236 004620              .WORD  L10073-.
6771 031240 004537 004064      JSR    R5,READ      ;GET T1LH FOR ERROR MESSAGE
6772 031244 120007              T1LH
6773 031246 002456              TMP7
6774 031250 103003              BCC    .+10         ;IF NO ERROR, PROCEED
6775 031252                      ERROR                   ;ELSE, REPORT IT

```



CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

6776 031252 104460
6777 031254
6778 031254 104410
6779 031256 004600
6780 031260 004537 004064
6781 031264 120013
6782 031266 002466
6783 031270 103003
6784 031272
6785 031272 104460
6786 031274
6787 031274 104410
6788 031276 004560
6789 031300
6790
6791 031300 104455
6792 031302 000062
6793 031304 016067
6794 031306 010762
6795 031310
6796 031310 012746 013174
6797 031314 012746 000001
6798 031320 010600
6799 031322 104415
6800 031324 062706 000004
6801
6802
6803
6804 031330 112737 000002 002453
6805 031336 004537 004310
6806 031342 120005
6807 031344 002453
6808 031346 103003
6809 031350
6810 031350 104460
6811 031352
6812 031352 104410
6813 031354 004502
6814 031356 004737 036112
6815 031362 102002
6816 031364
6817 031364 104410
6818 031366 001770
6819 031370 103036
6820 031372 004537 004064
6821 031376 120004
6822 031400 002450
6823 031402 103003
6824 031404
6825 031404 104460
6826 031406
6827 031406 104410
6828 031410 004446
6829 031412 004537 004064
6830 031416 120005
6831 031420 002452

```

```

ESCAPE TST ; AND EXIT THIS TEST TRAP C$ERROR
; .WORD C$ESCAPE
; L10073-.
JSR R5,READ ;GET ACR FOR ERROR MESSAGE
ACR
TMPB
BCC .+10 ;IF NO ERROR, PROCEED
ERROR ;ELSE, REPORT IT
ESCAPE TST ; AND EXIT THIS TEST TRAP C$ERROR
; .WORD C$ESCAPE
; L10073-.
GEDF EM50A,ERR50 ;YES, REPORT IT'S NOT BEING CLEARED @ INIT.
; 'DEVICE FATAL' ERROR # 50
TRAP C$ERDF
; .WORD 50
; .WORD EM50A
; .WORD ERR50
PRINTX #FMT50M ; & SAY THE COUNTERS HAVEN'T BEEN LOADED YET!
MOV #FMT50M,-(SP)
MOV #1,-(SP)
MOV SP,R0
TRAP C$PNTX
ADD #4,SP
-----
MOV #2,TMP5+1
JSR R5,WRITE ;INIT TIMER # 1 BY WRITING INTO
T1CH ;T1C-H (ADDR 05)
TMP5+1
BCC .+10 ;IF NO ERROR, PROCEED
ERROR ;ELSE, REPORT IT
ESCAPE TST ; AND EXIT THIS TEST TRAP C$ERROR
; .WORD C$ESCAPE
; L10073-.
JSR PC,GETT1 ;IS 'T1' SET?
BVC .+6 ;IF NO ERROR, PROCEED
ESCAPE SUB ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
TRAP C$ESCAPE
; .WORD L10074-.
BCC 6$ ;NO, GOOD.
JSR R5,READ ;GET T1CL FOR ERROR MESSAGE
T1CL
TMP4
BCC .+10 ;IF NO ERROR, PROCEED
ERROR ;ELSE, REPORT IT
ESCAPE TST ; AND EXIT THIS TEST TRAP C$ERROR
; .WORD C$ESCAPE
; L10073-.
JSR R5,READ ;GET T1CH FOR ERROR MESSAGE
T1CH
TMP5

```

CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

6832 031422 103003          BCC      .+10          ;IF NO ERROR, PROCEED
6833 031424          ERROR          ;ELSE, REPORT IT
6834 031424 104460          ESCAPE  TST          ;          AND EXIT THIS TEST          TRAP  C$ERROR
6835 031426          ;
6836 031426 104410          ;          AND EXIT THIS TEST          TRAP  C$ESCAPE
6837 031430 004426          .WORD  L10073-.
6838 031432 004537 004064  JSR      R5,READ          ;GET T1LH FOR ERROR MESSAGE
6839 031436 120007          T1LH
6840 031440 002456          TMP7
6841 031442 103003          BCC      .+10          ;IF NO ERROR, PROCEED
6842 031444          ERROR          ;ELSE, REPORT IT
6843 031444 104460          ESCAPE  TST          ;          AND EXIT THIS TEST          TRAP  C$ERROR
6844 031446          ;
6845 031446 104410          ;          AND EXIT THIS TEST          TRAP  C$ESCAPE
6846 031450 004406          .WORD  L10073-.
6847 031452          GEDF   EM50B,ERR50      ;YES, REPORT IT'S NOT BEING CLEARED @ INIT.
6848          ;          'DEVICE FATAL' ERROR # 51
6849 031452 104455          ;
6850 031454 000063          TRAP  C$ERDF
6851 031456 016135          .WORD  51
6852 031460 010762          .WORD  EM50B
6853 031462          .WORD  ERR50
6854 031462 104410          ESCAPE  SUB          ;AND EXIT SUBTEST          TRAP  C$ESCAPE
6855 031464 001672          .WORD  L10074-.
6856
6857
6858 ;-----;
6859 031466 004537 004064  6$:  JSR      R5,READ          ;GET ACR FOR LATER ERROR MESSAGES
6860 031472 120013          ACR
6861 031474 002466          TMPB
6862 031476 103003          BCC      .+10          ;IF NO ERROR, PROCEED
6863 031500          ERROR          ;ELSE, REPORT IT
6864 031500 104460          ESCAPE  TST          ;          AND EXIT THIS TEST          TRAP  C$ERROR
6865 031502          ;
6866 031502 104410          ;          AND EXIT THIS TEST          TRAP  C$ESCAPE
6867 031504 004352          .WORD  L10073-.
6868 031506 112737 000377 002445  MOVB   #377,TMP2+1      ;INITIALIZE ORB FOR INPUT/OUTPUT
6869 031514 004537 004310  JSR      R5,WRITE
6870 031520 120002          DDRB
6871 031522 002445          TMP2+1
6872 031524 103003          BCC      .+10          ;IF NO ERROR, PROCEED
6873 031526          ERROR          ;ELSE, REPORT IT
6874 031526 104460          ESCAPE  TST          ;          AND EXIT THIS TEST          TRAP  C$ERROR
6875 031530          ;
6876 031530 104410          ;          AND EXIT THIS TEST          TRAP  C$ESCAPE
6877 031532 004324          .WORD  L10073-.
6878 031534 112737 000377 002441  MOVB   #377,TMP0+1      ;SETUP VALUE FOR ORB
6879 031542 004537 004310  JSR      R5,WRITE          ;DO IT
6880 031546 120000          ORB
6881 031550 002441          TMP0+1
6882 031552 103003          BCC      .+10          ;IF NO ERROR, PROCEED
6883 031554          ERROR          ;ELSE, REPORT IT
6884 031554 104460          ESCAPE  TST          ;          AND EXIT THIS TEST          TRAP  C$ERROR
6885 031556          ;
6886 031556 104410          ;          AND EXIT THIS TEST          TRAP  C$ESCAPE
6887 031560 004276          .WORD  L10073-.

```



CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

6888 031562 004537 036060
6889 031566 252
6890 031567 252
6891
6892
6893
6894 031570 004737 036276
6895 031574 102002
6896 031576
6897 031576 104410
6898 031600 001556
6899 031602 103404
6900 031604
6901
6902 031604 104455
6903 031606 000064
6904 031610 017305
6905 031612 005276
6906 031614 004737 036150
6907
6908
6909 031620 103003
6910 031622
6911 031622 104460
6912 031624
6913 031624 104410
6914 031626 004230
6915 031630 000240
6916 031632 000240
6917 031634 000240
6918 031636 004537 004064
6919 031642 120004
6920 031644 002450
6921 031646 103003
6922 031650
6923 031650 104460
6924 031652
6925 031652 104410
6926 031654 004202
6927 031656 123737 002450 031566
6928 031664 001024
6929 031666 004537 004064
6930 031672 120005
6931 031674 002452
6932 031676 103003
6933 031700
6934 031700 104460
6935 031702
6936 031702 104410
6937 031704 004152
6938 031706 004537 004064
6939 031712 120006
6940 031714 002454
6941 031716 103003
6942 031720
6943 031720 104460

```

```

;LOAD TIMER # 1
7$: JSR R5,LODT1C
;BYTE 252
8$: .BYTE 252
-----
JSR PC,GETPB7 ;GET 'PB7'. IS IT SET?
BVC .+6 ;IF NO ERROR, PROCEED
ESCAPE SUB ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
;IT IS. (SET BY LOADING DDRB & ORB ABOVE)
BCS 9$ ;IF NOT, TIMER 1 CLEARED IT!!!
GEDF EM50W,ERR1 ; 'DEVICE FATAL' ERROR # 52
;IT TAKES SO MUCH TIME TO CHECK FOR (& MAYBE
; EVEN REPORT) THIS ERROR THAT WE SHOULD
; KICK OFF THE TIMER AGAIN
;IF NO ERROR, PROCEED
;ELSE, REPORT IT
; AND EXIT THIS TEST
;NOW WAIT A LITTLE WHILE -- LET T1CL CHANGE
;READ THE LOW COUNTER
;IF NO ERROR, PROCEED
;ELSE, REPORT IT
; AND EXIT THIS TEST
;MAKE SURE THE COUNTER IS DECREMENTING
;IT IS, NOW SEE IF THE HIGH COUNTER IS TOO
;GET T1CH FOR ERROR MESSAGE
;IF NO ERROR, PROCEED
;ELSE, REPORT IT
; AND EXIT THIS TEST
;GET T1LL FOR ERROR MESSAGE
;IF NO ERROR, PROCEED
;ELSE, REPORT IT

```

```

TRAP C$ESCAPE
.WORD L10074-.
TRAP C$ERDF
.WORD 52
.WORD EM50W
.WORD ERR1
TRAP C$ERROR
TRAP C$ESCAPE
.WORD L10073-.
TRAP C$ERROR
TRAP C$ESCAPE
.WORD L10073-.
TRAP C$ERROR
TRAP C$ESCAPE
.WORD L10073-.
TRAP C$ERROR

```

CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

6944 031722          ESCAPE TST          ;          AND EXIT THIS TEST          TRAP          C$ESCAPE
6945 031722 104410          ;          ;          .WORD          L10073-.
6946 031724 004132          GEDF      EM50D,ERR50      ;IT WASN'T -- REPORT THE ERROR      ;          'DEVICE FATAL' ERROR # 53      TRAP          C$ERDF
6947 031726          ;          ;          ;          ;          .WORD          53
6948 031726          ;          ;          ;          ;          .WORD          EM50D
6949 031726 104455          ;          ;          ;          ;          .WORD          ERR50
6950 031730 000065          12$:      MOV      #100,R3          ;INIT. TIMEOUT VALUE
6951 031732 016251          13$:      JSR      R5,READ          ;READ THE HIGH COUNTER
6952 031734 010762          T1CH
6953 031736 012703 000100          T1CH
6954 031742 004537 004064          TMP5
6955 031746 120005          BCC      .+10          ;IF NO ERROR, PROCEED
6956 031750 002452          ERROR          ;ELSE, REPORT IT
6957 031752 103003          ESCAPE TST          ;          AND EXIT THIS TEST          TRAP          C$ERROR
6958 031754          ;          ;          ;          ;          TRAP          C$ESCAPE
6959 031754 104460          ;          ;          ;          ;          .WORD          L10073-.
6960 031756          CMPB      TMP5,R3          ;DID IT CHANGE FROM THE LOADED VALUE?
6961 031756 104410          BNE      17$          ;YES, PROCEED WITH TESTING
6962 031760 004076          SOB      R3,13$          ;NO, IF NO TIMEOUT, TRY AGAIN
6963 031762 123737 002452 031567          JSR      R5,READ          ;GET IFR FOR ERROR MESSAGE
6964 031770 001037          IFR
6965 031772 077315          TMPD
6966 031774 004537 004064          BCC      .+10          ;IF NO ERROR, PROCEED
6967 032000 120015          ERROR          ;ELSE, REPORT IT
6968 032002 002472          ESCAPE TST          ;          AND EXIT THIS TEST          TRAP          C$ERROR
6969 032004 103003          ;          ;          ;          ;          TRAP          C$ESCAPE
6970 032006          ;          ;          ;          ;          .WORD          L10073-.
6971 032006 104460          JSR      R5,READ          ;GET T1LL FOR ERROR MESSAGE
6972 032010          T1LL
6973 032010 104410          TMP6
6974 032012 004044          BCC      .+10          ;IF NO ERROR, PROCEED
6975 032014 004537 004064          ERROR          ;ELSE, REPORT IT
6976 032020 120006          ESCAPE TST          ;          AND EXIT THIS TEST          TRAP          C$ERROR
6977 032022 002454          ;          ;          ;          ;          TRAP          C$ESCAPE
6978 032024 103003          ;          ;          ;          ;          .WORD          L10073-.
6979 032026          JSR      R5,READ          ;GET T1LH FOR ERROR MESSAGE
6980 032026 104460          T1LH
6981 032030          TMP7
6982 032030 104410          BCC      .+10          ;IF NO ERROR, PROCEED
6983 032032 004024          ERROR          ;ELSE, REPORT IT
6984 032034 004537 004064          ESCAPE TST          ;          AND EXIT THIS TEST          TRAP          C$ERROR
6985 032040 120007          ;          ;          ;          ;          TRAP          C$ESCAPE
6986 032042 002456          ;          ;          ;          ;          .WORD          L10073-.
6987 032044 103003          JSR      R5,READ          ;GET T1LH FOR ERROR MESSAGE
6988 032046          T1LH
6989 032046 104460          TMP7
6990 032050          BCC      .+10          ;IF NO ERROR, PROCEED
6991 032050 104410          ERROR          ;ELSE, REPORT IT
6992 032052 004004          ESCAPE TST          ;          AND EXIT THIS TEST          TRAP          C$ERROR
6993 032054          ;          ;          ;          ;          TRAP          C$ESCAPE
6994 032054          ;          ;          ;          ;          .WORD          L10073-.
6995 032054 104455          GEDF      EM50E,ERR50      ;ELSE, REPORT THAT HIGH COUNTER ISN'T RUNNING
6996 032056 000066          ;          ;          ;          ;          ;          'DEVICE FATAL' ERROR # 54      TRAP          C$ERDF
6997 032060 016305          ;          ;          ;          ;          ;          ;          .WORD          54
6998 032062 010762          ;          ;          ;          ;          ;          .WORD          EM50E
6999 032064          ESCAPE SUB          ;IN THAT CASE, WE CAN'T PROCEED WITH TESTING EITHER          .WORD          ERR50

```



CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

7000 032064 104410
7001 032066 001270
7002
7003
-----
7004 032070 112737 000377 002445 17$:  MOVB  #377,TMP2+1 ;INITIALIZE ORB FOR INPUT/OUTPUT
7005 032076 004537 004310 JSR  R5,WRITE
7006 032102 120002 DDRB
7007 032104 002445 TMP2+1
7008 032106 103003 BCC  .+10 ;IF NO ERROR, PROCEED
7009 032110 ERROR ;ELSE, REPORT IT
7010 032110 104460 ESCAPE TST ; AND EXIT THIS TEST TRAP CSERROR
7011 032112
7012 032112 104410
7013 032114 003742
7014 032116 023737 002440 002441 CMP  TMP0,TMP0+1 ;CLEAR PB7 BY WRITING INTO ORB TRAP CSUCCESS
7015 032124 004537 004322 JSR  R5,WRITEI ;.WORD L10073-.
7016 032130 120000 ORB
7017 032132 000030 30 ; (THIS CLEARS DTR & RTS! ALSO)
7018 032134 103003 BCC  .+10 ;IF NO ERROR, PROCEED
7019 032136 ERROR ;ELSE, REPORT IT
7020 032136 104460 ESCAPE TST ; AND EXIT THIS TEST TRAP CSERROR
7021 032140
7022 032140 104410
7023 032142 003714
7024 032144 004537 036060 JSR  R5,LODT1C ;RE-LOAD TIMER # 1 WITH A VALUE WHICH CAUSE AN
7025 032150 001 18$: .BYTE 1 ;ALMOST IMMEDIATE TIMEOUT
7026 032151 000 19$: .BYTE 0 ; (ADDRESS OF HIGH BYTE FOR T1C-H (ADDR 05))
-----
7027
7028
7029 032152 004737 036112 JSR  PC,GETT1 ;WAS 'T1' SET BY THE ABOVE OPERATION?
7030 032156 102002 BVC  .+6 ;IF NO ERROR, PROCEED
7031 032160 ESCAPE SUB ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT TRAP CSUCCESS
7032 032160 104410
7033 032162 001174
7034 032164 103426
7035 032166 004537 004064 BCS  20$ ;YES, OK --- CONTINUE ERROR CHECKING
7036 032172 120006 JSR  R5,READ ;GET T1LL FOR ERROR MESSAGE
7037 032174 002454 T1LL
7038 032176 103003 TMP6
7039 032200 BCC  .+10 ;IF NO ERROR, PROCEED
7040 032200 104460 ERROR ;ELSE, REPORT IT TRAP CSERROR
7041 032202 ESCAPE TST ; AND EXIT THIS TEST
7042 032202 104410
7043 032204 003652
7044 032206 004537 004064 JSR  R5,READ ;GET T1LH FOR ERROR MESSAGE TRAP CSUCCESS
7045 032212 120007 T1LH ;.WORD L10073-.
7046 032214 002456 TMP7
7047 032216 103003 BCC  .+10 ;IF NO ERROR, PROCEED
7048 032220 ERROR ;ELSE, REPORT IT TRAP CSERROR
7049 032220 104460 ESCAPE TST ; AND EXIT THIS TEST
7050 032222
7051 032222 104410
7052 032224 003632
7053 032226 GEDF  EM50F,ERR50 ;NO BAD NEWS! REPORT THE FAILURE
7054 ; 'DEVICE FATAL' ERROR # 55
7055 032226 104455 TRAP CSERDF

```

CVOMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

7056	032230	000067					.WORD	55
7057	032232	016341					.WORD	EM50F
7058	032234	010762					.WORD	ERR50
7059	032236			ESCAPE	SUB			; AND GET OUT OF SUBTEST
7060	032236	104410					TRAP	C\$ESCAPE
7061	032240	001116					.WORD	L10074-
7062	032242	004737	036276	20\$:	JSR	PC,GETPB7		;GET 'PB7'. IS IT CLEARED?
7063	032246	102002			BVC	+.6		;IF NO ERROR, PROCEED
7064	032250				ESCAPE	SUB		;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
7065	032250	104410					TRAP	C\$ESCAPE
7066	032252	001104					.WORD	L10074-
7067	032254	103024			BCC	40\$		;IF CLEARED, DDRB IS STILL IN CONTROL OF IT
7068	032256	004537	004064		JSR	R5,READ		;GET T1LL FOR ERROR MESSAGE
7069	032262	120006			T1LL			
7070	032264	002454			TMP6			
7071	032266	103003			BCC	+.10		;IF NO ERROR, PROCEED
7072	032270				ERROR			;ELSE, REPORT IT
7073	032270	104460					TRAP	C\$ERROR
7074	032272				ESCAPE	TST		; AND EXIT THIS TEST
7075	032272	104410					TRAP	C\$ESCAPE
7076	032274	003562					.WORD	L10073-
7077	032276	004537	004064		JSR	R5,READ		;GET T1LH FOR ERROR MESSAGE
7078	032302	120007			T1LH			
7079	032304	002456			TMP7			
7080	032306	103003			BCC	+.10		;IF NO ERROR, PROCEED
7081	032310				ERROR			;ELSE, REPORT IT
7082	032310	104460					TRAP	C\$ERROR
7083	032312				ESCAPE	TST		; AND EXIT THIS TEST
7084	032312	104410					TRAP	C\$ESCAPE
7085	032314	003542					.WORD	L10073-
7086	032316				GEDF	EM50W,ERR50		;ELSE, IT'S BEING SET BY TIMER 1 IN MODE 0!
7087								; 'DEVICE FATAL' ERROR # 56
7088	032316	104455					TRAP	C\$ERDF
7089	032320	000070					.WORD	56
7090	032322	017305					.WORD	EM50W
7091	032324	010762					.WORD	ERR50
7092	032326	004537	004064	40\$:	JSR	R5,READ		;READ T1C-H (ADDR 05) TO SEE IF THIS CLEARS 'T1'
7093	032332	120005			T1CH			;(THIS VALUE ISN'T CHECKED BECAUSE IT CAN BE
7094	032334	002452			TMP5			; ALMOST ANYTHING)
7095	032336	103003			BCC	+.10		;IF NO ERROR, PROCEED
7096	032340				ERROR			;ELSE, REPORT IT
7097	032340	104460					TRAP	C\$ERROR
7098	032342				ESCAPE	TST		; AND EXIT THIS TEST
7099	032342	104410					TRAP	C\$ESCAPE
7100	032344	003512					.WORD	L10073-
7101	032346	004737	036112		JSR	PC,GETT1		;PUT THE CURRENT 'T1' VALUE INTO THE CARRY BIT
7102	032352	102002			BVC	+.6		;IF NO ERROR, PROCEED
7103	032354				ESCAPE	SUB		;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
7104	032354	104410					TRAP	C\$ESCAPE
7105	032356	001000					.WORD	L10074-
7106	032360	103425			BCS	21\$		;IF SET, READING T1CH DIDN'T CLEAR IT -- OK!
7107	032362	004537	004064		JSR	R5,READ		;GET T1LL FOR ERROR MESSAGE
7108	032366	120006			T1LL			
7109	032370	002454			TMP6			
7110	032372	103003			BCC	+.10		;IF NO ERROR, PROCEED
7111	032374				ERROR			;ELSE, REPORT IT



CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

7112 032374 104460
7113 032376 104410          ESCAPE TST          ; AND EXIT THIS TEST          TRAP C$ERROR
7114 032376 104410          ;                               ;                               TRAP C$ESCAPE
7115 032400 003456          ;                               ;                               .WORD L10073-.
7116 032402 004537 004064 JSR R5,READ          ;GET T1LH FOR ERROR MESSAGE
7117 032406 120007          T1LH
7118 032410 002456          TMP7
7119 032412 103003          BCC .+10          ;IF NO ERROR, PROCEED
7120 032414 104460          ERROR          ;ELSE, REPORT IT
7121 032414 104460          ESCAPE TST          ; AND EXIT THIS TEST          TRAP C$ERROR
7122 032416 104410          ;                               ;                               TRAP C$ESCAPE
7123 032416 104410          ;                               ;                               .WORD L10073-.
7124 032420 003436          GEDF EM50G,ERR50    ;IF CLEARED! BAD VIA CHIP!
7125 032422 104455          ; 'DEVICE FATAL' ERROR # 57
7126 032422 104455          ;                               ;                               TRAP C$ERDF
7127 032422 104455          ;                               ;                               .WORD 57
7128 032424 000071          ;                               ;                               .WORD EM50G
7129 032426 016406          ;                               ;                               .WORD ERR50
7130 032430 010762          ;                               ;                               .WORD
7131 032432 000507          BR 28$          ;BYPASS THE REST OF THIS SECTION OF TESTING
7132 032432 000507
7133 032434 004537 004064 21$: JSR R5,READ          ;READ T1L-L (ADDR 06)
7134 032440 120006          T1LL
7135 032442 002454          TMP6          ;THIS SHOULD RETURN A 001
7136 032444 103003          BCC .+10          ;IF NO ERROR, PROCEED
7137 032446 104460          ERROR          ;ELSE, REPORT IT
7138 032446 104460          ESCAPE TST          ; AND EXIT THIS TEST          TRAP C$ERROR
7139 032450 104410          ;                               ;                               TRAP C$ESCAPE
7140 032450 104410          ;                               ;                               .WORD L10073-.
7141 032452 003404          CMPB TMP6,18$    ;CHECK T1L-L (ADDR 06) AGAINST LOADED VALUE
7142 032454 123737 002454 032150 BEQ 23$          ;IF SAME, PROCEED
7143 032462 001415          JSR R5,READ          ;GET T1LH FOR ERROR MESSAGE
7144 032464 004537 004064 JSR R5,READ
7145 032470 120007          T1LH
7146 032472 002456          TMP7
7147 032474 103003          BCC .+10          ;IF NO ERROR, PROCEED
7148 032476 104460          ERROR          ;ELSE, REPORT IT
7149 032476 104460          ESCAPE TST          ; AND EXIT THIS TEST          TRAP C$ERROR
7150 032500 104410          ;                               ;                               TRAP C$ESCAPE
7151 032500 104410          ;                               ;                               .WORD L10073-.
7152 032502 003354          GEDF EM50H,ERR50  ;ELSE, REPORT BAD LOAD OF T1L-L (ADDR 06)
7153 032504 104455          ; 'DEVICE FATAL' ERROR # 58
7154 032504 104455          ;                               ;                               TRAP C$ERDF
7155 032504 104455          ;                               ;                               .WORD 58
7156 032506 000072          ;                               ;                               .WORD EM50H
7157 032510 016450          ;                               ;                               .WORD ERR50
7158 032512 010762          ;                               ;                               .WORD
7159 032514 000456          BR 28$          ;BYPASS THE REST OF THIS SECTION OF TESTING
7160 032514 000456
7161 032516 004737 036112 23$: JSR PC,GETT1        ;IS 'T1' STILL SET?
7162 032522 102002          BVC .+6          ;IF NO ERROR, PROCEED
7163 032524 104410          ESCAPE SUB          ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
7164 032524 104410          ;                               ;                               TRAP C$ESCAPE
7165 032526 000630          ;                               ;                               .WORD L10074-.
7166 032530 103415          BCS 24$          ;YES, ALL'S OK
7167 032532 004537 004064 JSR R5,READ          ;GET T1LH FOR ERROR MESSAGE

```

CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

7168 032536 120007      T1LH
7169 032540 002456      TMP7
7170 032542 103003      BCC      .+10      ;IF NO ERROR, PROCEED
7171 032544              ERROR      ;ELSE, REPORT IT
7172 032544 104460              ESCAPE TST      ;           AND EXIT THIS TEST      TRAP      C$ERROR
7173 032546              ESCAPE TST      ;           AND EXIT THIS TEST      TRAP      C$ESCAPE
7174 032546 104410              ESCAPE TST      ;           AND EXIT THIS TEST      .WORD    L10073-.
7175 032550 003306              GEDF      EM50I,ERR50      ;NO!  BAD VIA CHIP!
7176 032552              GEDF      EM50I,ERR50      ;           'DEVICE FATAL' ERROR # 59
7177 032552              GEDF      EM50I,ERR50      ;           'DEVICE FATAL' ERROR # 59
7178 032552 104455              GEDF      EM50I,ERR50      ;           'DEVICE FATAL' ERROR # 59      TRAP      C$ERDF
7179 032554 000073              GEDF      EM50I,ERR50      ;           'DEVICE FATAL' ERROR # 59      .WORD    59
7180 032556 016536              GEDF      EM50I,ERR50      ;           'DEVICE FATAL' ERROR # 59      .WORD    EM50I
7181 032560 010762              GEDF      EM50I,ERR50      ;           'DEVICE FATAL' ERROR # 59      .WORD    ERR50
7182 032562 000433              BR      28$      ;BYPASS THE REST OF THIS SECTION OF TESTING
7183 032564 004537 004064      24$:  JSR      R5,READ      ;READ T1L-H (ADDR 07)
7184 032564 004537 004064      24$:  JSR      R5,READ      ;READ T1L-H (ADDR 07)
7185 032570 120007      T1LH
7186 032572 002456      TMP7
7187 032574 103003      BCC      .+10      ;THIS SHOULD RETURN A 000
7188 032576              ERROR      ;IF NO ERROR, PROCEED
7189 032576 104460              ESCAPE TST      ;           AND EXIT THIS TEST      TRAP      C$ERROR
7190 032600              ESCAPE TST      ;           AND EXIT THIS TEST      TRAP      C$ESCAPE
7191 032600 104410              ESCAPE TST      ;           AND EXIT THIS TEST      .WORD    L10073-.
7192 032602 003254      CMPB      TMP7,19$      ;CHECK T1L-H (ADDR 07) AGAINST LOADED VALUE
7193 032604 123737 002456 032151      BEQ      26$      ;IF SAME, PROCEED
7194 032612 001405      GEDF      EM50J,ERR50      ;ELSE, REPORT BAD LOAD OF T1L-H (ADDR 07)
7195 032614              GEDF      EM50J,ERR50      ;           'DEVICE FATAL' ERROR # 60
7196 032614              GEDF      EM50J,ERR50      ;           'DEVICE FATAL' ERROR # 60
7197 032614 104455              GEDF      EM50J,ERR50      ;           'DEVICE FATAL' ERROR # 60      TRAP      C$ERDF
7198 032616 000074              GEDF      EM50J,ERR50      ;           'DEVICE FATAL' ERROR # 60      .WORD    60
7199 032620 016600              GEDF      EM50J,ERR50      ;           'DEVICE FATAL' ERROR # 60      .WORD    EM50J
7200 032622 010762              GEDF      EM50J,ERR50      ;           'DEVICE FATAL' ERROR # 60      .WORD    ERR50
7201 032624 000412              BR      28$      ;BYPASS THE REST OF THIS SECTION OF TESTING
7202 032626 004737 036112      26$:  JSR      PC,GETT1      ;IS 'T1' STILL SET?
7203 032632 102002      BVC      .+6      ;IF NO ERROR, PROCEED
7204 032634              ESCAPE SUB      ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
7205 032634 104410              ESCAPE SUB      ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT      TRAP      C$ESCAPE
7206 032634 104410              ESCAPE SUB      ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT      .WORD    L10074-.
7207 032636 000520              BCS      28$      ;YES, ALL'S OK
7208 032640 103404              GEDF      EM50K,ERR50      ;NO!  BAD VIA CHIP!
7209 032642              GEDF      EM50K,ERR50      ;           'DEVICE FATAL' ERROR # 61
7210 032642              GEDF      EM50K,ERR50      ;           'DEVICE FATAL' ERROR # 61
7211 032642 104455              GEDF      EM50K,ERR50      ;           'DEVICE FATAL' ERROR # 61      TRAP      C$ERDF
7212 032644 000075              GEDF      EM50K,ERR50      ;           'DEVICE FATAL' ERROR # 61      .WORD    61
7213 032646 016666              GEDF      EM50K,ERR50      ;           'DEVICE FATAL' ERROR # 61      .WORD    EM50K
7214 032650 010762              GEDF      EM50K,ERR50      ;           'DEVICE FATAL' ERROR # 61      .WORD    ERR50
7215 032652 004537 004064      28$:  JSR      R5,READ      ;READ T1C-L (ADDR 04)
7216 032652 004537 004064      28$:  JSR      R5,READ      ;READ T1C-L (ADDR 04)
7217 032656 120004      T1CL      ;(THIS VALUE ISN'T CHECKED BECAUSE IT CAN BE
7218 032660 002450      TMP4      ;ALMOST ANYTHING)
7219 032662 103003      BCC      .+10      ;IF NO ERROR, PROCEED
7220 032664              ERROR      ;ELSE, REPORT IT
7221 032664 104460              ERROR      ;ELSE, REPORT IT      TRAP      C$ERROR
7222 032664 104460              ERROR      ;ELSE, REPORT IT
7223 032664 104460              ERROR      ;ELSE, REPORT IT

```



CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

7224 032666          ESCAPE TST          ; AND EXIT THIS TEST
7225 032666 104410          ; TRAP C$ESCAPE
7226 032670 003166          ; .WORD L10073-.
7227 032672 004737 036112  JSR PC,GETT1      ; IS 'T1' CLEARED NOW
7228 032676 102002          BVC .+6           ; IF NO ERROR, PROCEED
7229 032700          ESCAPE SUB           ; ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
7230 032700 104410          ; TRAP C$ESCAPE
7231 032702 000454          ; .WORD L10074-.
7232 032704 103024          BCC 29$           ; YES, ALL'S OK
7233 032706 004537 004064  JSR R5,READ       ; GET T1LL FOR ERROR MESSAGE
7234 032712 120006          T1LL
7235 032714 002454          TMP6
7236 032716 103003          BCC .+10         ; IF NO ERROR, PROCEED
7237 032720          ERROR                ; ELSE, REPORT IT
7238 032720 104460          ; TRAP C$ERROR
7239 032722          ESCAPE TST          ; AND EXIT THIS TEST
7240 032722 104410          ; TRAP C$ESCAPE
7241 032724 003132          ; .WORD L10073-.
7242 032726 004537 004064  JSR R5,READ       ; GET T1LH FOR ERROR MESSAGE
7243 032732 120007          T1LH
7244 032734 002456          TMP7
7245 032736 103003          BCC .+10         ; IF NO ERROR, PROCEED
7246 032740          ERROR                ; ELSE, REPORT IT
7247 032740 104460          ; TRAP C$ERROR
7248 032742          ESCAPE TST          ; AND EXIT THIS TEST
7249 032742 104410          ; TRAP C$ESCAPE
7250 032744 003112          ; .WORD L10073-.
7251 032746          GEDF EM50C,ERR50     ; NO! BAD VIA CHIP!
7252          ; 'DEVICE FATAL' ERROR # 62
7253 032746 104455          ; TRAP C$ERDF
7254 032750 000076          ; .WORD 62
7255 032752 016203          ; .WORD EM50C
7256 032754 010762          ; .WORD ERR50
7257
7258 ;-----
7259
7260 032756 004537 004310  29$: JSR R5,WRITE   ; RE-WRITE INTO T1C-H (ADDR 05) TO SET T1 AGAIN
7261 032762 120005          T1CH
7262 032764 002453          TMP5+1
7263 032766 103003          BCC .+10         ; IF NO ERROR, PROCEED
7264 032770          ERROR                ; ELSE, REPORT IT
7265 032770 104460          ; TRAP C$ERROR
7266 032772          ESCAPE TST          ; AND EXIT THIS TEST
7267 032772 104410          ; TRAP C$ESCAPE
7268 032774 003062          ; .WORD L10073-.
7269 032776 004737 036112  JSR PC,GETT1      ; IS 'T1' SET AGAIN
7270 033002 102002          BVC .+6           ; IF NO ERROR, PROCEED
7271 033004          ESCAPE SUB           ; ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
7272 033004 104410          ; TRAP C$ESCAPE
7273 033006 000350          ; .WORD L10074-.
7274 033010 103426          BCS 32$           ; YES, ALL'S WELL (AGAIN?)
7275 033012 004537 004064  JSR R5,READ       ; GET T1LL FOR ERROR MESSAGE
7276 033016 120006          T1LL
7277 033020 002454          TMP6
7278 033022 103003          BCC .+10         ; IF NO ERROR, PROCEED
7279 033024          ERROR                ; ELSE, REPORT IT

```

CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

7280 033024 104460
7281 033026          ESCAPE TST          :          AND EXIT THIS TEST          TRAP  CSERROR
7282 033026 104410
7283 033030 003026          .WORD  CS_ESCAPE
7284 033032 004537 004064 JSR    R5,READ          :GET T1LH FOR ERROR MESSAGE          .WORD  L10073-.
7285 033036 120007
7286 033040 002456          T1LH
7287 033042 103003          TMP7
7288 033044          BCC    .+10          :IF NO ERROR, PROCEED
7289 033044 104460          ERROR          :ELSE, REPORT IT
7290 033046          ESCAPE TST          :          AND EXIT THIS TEST          TRAP  CSERROR
7291 033046 104410
7292 033050 003006          .WORD  CS_ESCAPE
7293 033052          GEDF   EM50L,ERR50    :NO!  SOMETHING WENT WRONG! REPORT IT
7294          :          'DEVICE FATAL' ERROR # 63          .WORD  L10073-.
7295 033052 104455          TRAP  CSERDF
7296 033054 000077          .WORD  63
7297 033056 016730          .WORD  EM50L
7298 033060 010762          .WORD  ERR50
7299 033062          ESCAPE SUB          :          AND EXIT FROM THIS SUBTEST    TRAP  CS_ESCAPE
7300 033062 104410          .WORD  L10074-.
7301 033064 000272
7302
7303 -----
7304
7305 033066 112737 000125 002455 32$:  MOVB  #125,TMP6+1    :USING A DIFFERENT VALUE -- 55 HEX.,
7306 033074 004537 004310 JSR    R5,WRITE    :RE-LOAD T1L-L (ADDR 06)
7307 033100 120006          T1LL
7308 033102 002455          TMP6+1
7309 033104 103003          BCC    .+10          :IF NO ERROR, PROCEED
7310 033106          ERROR          :ELSE, REPORT IT
7311 033106 104460          ESCAPE TST          :          AND EXIT THIS TEST          TRAP  CSERROR
7312 033110
7313 033110 104410          .WORD  CS_ESCAPE
7314 033112 002744          .WORD  L10073-.
7315 033114 004737 036112 JSR    PC,GETT1    :IS 'T1' STILL SET?
7316 033120 102002          BVC    .+6          :IF NO ERROR, PROCEED
7317 033122          ESCAPE SUB          :ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
7318 033122 104410          TRAP  CS_ESCAPE
7319 033124 000232          .WORD  L10074-.
7320 033126 103426          BCS    33$          :YES, ALL'S STILL OK
7321 033130 004537 004064 JSR    R5,READ    :GET T1LL FOR ERROR MESSAGE
7322 033134 120006          T1LL
7323 033136 002454          TMP6
7324 033140 103003          BCC    .+10          :IF NO ERROR, PROCEED
7325 033142          ERROR          :ELSE, REPORT IT
7326 033142 104460          ESCAPE TST          :          AND EXIT THIS TEST          TRAP  CSERROR
7327 033144
7328 033144 104410          .WORD  CS_ESCAPE
7329 033146 002710          .WORD  L10073-.
7330 033150 004537 004064 JSR    R5,READ    :GET T1LH FOR ERROR MESSAGE
7331 033154 120007          T1LH
7332 033156 002456          TMP7
7333 033160 103003          BCC    .+10          :IF NO ERROR, PROCEED
7334 033162          ERROR          :ELSE, REPORT IT
7335 033162 104460          TRAP  CSERROR

```



CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

7336 033164          ESCAPE TST          ;          AND EXIT THIS TEST          TRAP      C$ESCAPE
7337 033164 104410          ;          ;          .WORD      L10073-.
7338 033166 002670          ;          ;          ;          ;
7339 033170          GEDF   EM50M,ERR50      ;NO!  SOMETHING WENT WRONG!  REPORT IT
7340          ;          ;          ;          'DEVICE FATAL' ERROR # 64
7341 033170 104455          ;          ;          ;          ;          TRAP      C$ERDF
7342 033172 000100          ;          ;          ;          ;          .WORD      64
7343 033174 017012          ;          ;          ;          ;          .WORD      EM50M
7344 033176 010762          ;          ;          ;          ;          .WORD      ERR50
7345 033200          ESCAPE SUB          ;          AND EXIT FROM THIS SUBTEST
7346 033200 104410          ;          ;          ;          ;          TRAP      C$ESCAPE
7347 033202 000154          ;          ;          ;          ;          .WORD      L10074-.
7348
7349
7350          ;-----;
7351 033204 112737 000125 002453 33$:  MOVB   #125,TMP5+1      ;AND USING THE SAME VALUE AGAIN (55 HEX),
7352 033212 004537 004310          JSR    R5,WRITE        ;NOW LOAD T1C-H (ADDR 05)
7353 033216 120005          T1CH
7354 033220 002453          TMP5+1
7355 033222 103003          BCC   .+10            ;IF NO ERROR, PROCEED
7356 033224          ERROR                ;ELSE, REPORT IT
7357 033224 104460          ;          ;          ;          ;          TRAP      C$ERROR
7358 033226          ESCAPE TST          ;          AND EXIT THIS TEST
7359 033226 104410          ;          ;          ;          ;          TRAP      C$ESCAPE
7360 033230 002626          ;          ;          ;          ;          .WORD      L10073-.
7361 033232 004737 036112          JSR    PC,GETT1        ;'T1' SHOULD NOW BE CLEARED
7362 033236 102002          BVC   .+6            ;IF NO ERROR, PROCEED
7363 033240          ESCAPE SUB          ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
7364 033240 104410          ;          ;          ;          ;          TRAP      C$ESCAPE
7365 033242 000114          ;          ;          ;          ;          .WORD      L10074-.
7366 033244 103044          BCC   34$            ;IT WAS, ALL'S WELL THAT END'S WELL (I THINK!?)
7367 033246 004537 004064          JSR    R5,READ        ;GET T1CL FOR ERROR MESSAGE
7368 033252 120004          T1CL
7369 033254 002450          TMP4
7370 033256 103003          BCC   .+10            ;IF NO ERROR, PROCEED
7371 033260          ERROR                ;ELSE, REPORT IT
7372 033260 104460          ;          ;          ;          ;          TRAP      C$ERROR
7373 033262          ESCAPE TST          ;          AND EXIT THIS TEST
7374 033262 104410          ;          ;          ;          ;          TRAP      C$ESCAPE
7375 033264 002572          ;          ;          ;          ;          .WORD      L10073-.
7376 033266 004537 004064          JSR    R5,READ        ;GET T1CH FOR ERROR MESSAGE
7377 033272 120005          T1CH
7378 033274 002452          TMP5
7379 033276 103003          BCC   .+10            ;IF NO ERROR, PROCEED
7380 033300          ERROR                ;ELSE, REPORT IT
7381 033300 104460          ;          ;          ;          ;          TRAP      C$ERROR
7382 033302          ESCAPE TST          ;          AND EXIT THIS TEST
7383 033302 104410          ;          ;          ;          ;          TRAP      C$ESCAPE
7384 033304 002552          ;          ;          ;          ;          .WORD      L10073-.
7385 033306 004537 004064          JSR    R5,READ        ;GET T1LL FOR ERROR MESSAGE
7386 033312 120006          T1LL
7387 033314 002454          TMP6
7388 033316 103003          BCC   .+10            ;IF NO ERROR, PROCEED
7389 033320          ERROR                ;ELSE, REPORT IT
7390 033320 104460          ;          ;          ;          ;          TRAP      C$ERROR
7391 033322          ESCAPE TST          ;          AND EXIT THIS TEST

```

CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

7392 033322 104410                                TRAP  C$ESCAPE
7393 033324 002532                                .WORD L10073-.
7394 033326 004537 004064  JSR    R5,READ      ;GET T1LH FOR ERROR MESSAGE
7395 033332 120007                                T1LH
7396 033334 002456                                TMP7
7397 033336 103003  BCC    .+10        ;IF NO ERROR, PROCEED
7398 033340                                ERROR            ;ELSE, REPORT IT
7399 033340 104460                                TRAP  C$ERROR
7400 033342                                ESCAPE TST      ;      AND EXIT THIS TEST
7401 033342 104410                                TRAP  C$ESCAPE
7402 033344 002512                                .WORD L10073-.
7403 033346                                GEDF   EM50N,ERR50 ;IT WASN'T! SOMETHING WENT WRONG! REPORT IT
7404                                ;      'DEVICE FATAL' ERROR # 65
7405 033346 104455                                TRAP  C$ERDF
7406 033350 000101                                .WORD 65
7407 033352 017054                                .WORD EM50N
7408 033354 010762                                .WORD ERR50
7409
7410 033356                                34$:  ENDSUB
7411 033356                                L10074:
7412 033356 104403                                TRAP  C$ESUB
7413                                ;=====
7414
7415                                ; TEST TIMER # 1 USING ONE-SHOT MODE WITH OUTPUT ON PB7 ENABLED.
7416
7417 033360                                BGNSUB
7418 033360                                T27.2:
7419 033360 104402                                TRAP  C$BSUB
7420 033362 004737 003762  JSR    PC,MSTCLR  ;INIT DMV & ENTER M-LOOP
7421 033366 103003  BCC    1$        ;IF NO ERROR, PROCEED WITH TESTING
7422 033370                                ERROR            ;ELSE, REPORT ERROR
7423 033370 104460                                TRAP  C$ERROR
7424 033372                                ESCAPE TST      ;      & EXIT TEST
7425 033372 104410                                TRAP  C$ESCAPE
7426 033374 002462                                .WORD L10073-.
7427 033376 004537 004660  1$:  JSR    R5,INITT1 ;INITIALIZE TIMER # 1
7428 033402 000000                                ;      0 ==> LATCHES
7429 033404 000200                                ;      MODE 2 & 'T1' INT. ENABLE FLAG CLEARED
7430 033406 103003  BCC    .+10        ;IF NO ERROR, PROCEED
7431 033410                                ERROR            ;ELSE, REPORT IT
7432 033410 104460                                TRAP  C$ERROR
7433 033412                                ESCAPE TST      ;      AND EXIT THIS TEST
7434 033412 104410                                TRAP  C$ESCAPE
7435 033414 002442                                .WORD L10073-.
7436
7437                                ; MODE 2 IS ONE-SHOT MODE WITH OUTPUT ON PB7 CONTROLLED BY TIMER 1
7438
7439 033416 004737 036112  JSR    PC,GETT1  ;IS 'T1' SET?
7440 033422 102002  BVC    .+6        ;IF NO ERROR, PROCEED
7441 033424                                ESCAPE SUB      ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
7442 033424 104410                                TRAP  C$ESCAPE
7443 033426 002426                                .WORD L10075-.
7444 033430 103123  BCC    6$        ;NO, GOOD.
7445                                ;YES, REPORT IT'S NOT BEING CLEARED @ INIT.
7446 033432 004537 004064  JSR    R5,READ  ;GET ACR FOR ERROR MESSAGE
7447 033436 120013  ACR

```



CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

7448 033440 002466      TMPB
7449 033442 103003      BCC      .+10      ;IF NO ERROR, PROCEED
7450 033444              ERROR,      ;ELSE, REPORT IT
7451 033444 104460              ESCAPE TST      ;
7452 033446              ;      AND EXIT THIS TEST      TRAP      C$ERROR
7453 033446 104410              ;      TRAP      C$ESCAPE
7454 033450 002406              ;      .WORD      L10073-.
7455 033452 004537 004064      JSR      R5,READ      ;GET T1CL FOR ERROR MESSAGE
7456 033456 120004              T1CL
7457 033460 002450              TMP4
7458 033462 103003      BCC      .+10      ;IF NO ERROR, PROCEED
7459 033464              ERROR      ;ELSE, REPORT IT
7460 033464 104460              ESCAPE TST      ;
7461 033466              ;      AND EXIT THIS TEST      TRAP      C$ERROR
7462 033466 104410              ;      TRAP      C$ESCAPE
7463 033470 002366              ;      .WORD      L10073-.
7464 033472 004537 004064      JSR      R5,READ      ;GET T1CH FOR ERROR MESSAGE
7465 033476 120005              T1CH
7466 033500 002452              TMP5
7467 033502 103003      BCC      .+10      ;IF NO ERROR, PROCEED
7468 033504              ERROR      ;ELSE, REPORT IT
7469 033504 104460              ESCAPE TST      ;
7470 033506              ;      AND EXIT THIS TEST      TRAP      C$ERROR
7471 033506 104410              ;      TRAP      C$ESCAPE
7472 033510 002346              ;      .WORD      L10073-.
7473 033512 004537 004064      JSR      R5,READ      ;GET T1LL FOR ERROR MESSAGE
7474 033516 120006              T1LL
7475 033520 002454              TMP6
7476 033522 103003      BCC      .+10      ;IF NO ERROR, PROCEED
7477 033524              ERROR      ;ELSE, REPORT IT
7478 033524 104460              ESCAPE TST      ;
7479 033526              ;      AND EXIT THIS TEST      TRAP      C$ERROR
7480 033526 104410              ;      TRAP      C$ESCAPE
7481 033530 002326              ;      .WORD      L10073-.
7482 033532 004537 004064      JSR      R5,READ      ;GET T1LH FOR ERROR MESSAGE
7483 033536 120007              T1LH
7484 033540 002456              TMP7
7485 033542 103003      BCC      .+10      ;IF NO ERROR, PROCEED
7486 033544              ERROR      ;ELSE, REPORT IT
7487 033544 104460              ESCAPE TST      ;
7488 033546              ;      AND EXIT THIS TEST      TRAP      C$ERROR
7489 033546 104410              ;      TRAP      C$ESCAPE
7490 033550 002306              ;      .WORD      L10073-.
7491 033552              GEDF      EM50A,ERR50 ;REPORT 'T1' NOT CLEARED @ INIT.
7492              ;      'DEVICE FATAL' ERROR # 66
7493 033552 104455              ;      TRAP      C$ERDF
7494 033554 000102              ;      .WORD      66
7495 033556 016067              ;      .WORD      EM50A
7496 033560 010762              ;      .WORD      ERR50
7497
7498
7499
-----
7500 033562 112737 000002 002453      MOVB      #2,TMP5+1
7501 033570 004537 004310      JSR      R5,WRITE      ;INIT TIMER # 1 BY WRITING INTO
7502 033574 120005              T1CH      ;T1C-H (ADDR 05)
7503 033576 002453              TMP5+1

```

CVI:MAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

7504 033600 103003          BCC      .+10          ;IF NO ERROR, PROCEED
7505 033602                ERROR          ;ELSE, REPORT IT
7506 033602 104460                ESCAPE  TST          ;           AND EXIT THIS TEST          TRAP  C$ERROR
7507 033604                ESCAPE  TST          ;           AND EXIT THIS TEST          TRAP  C$ESCAPE
7508 033604 104410                ESCAPE  TST          ;           AND EXIT THIS TEST          .WORD L10073-.
7509 033606 002250                JSR      PC,GETT1      ;IS 'T1' SET?
7510 033610 004737 036112        BVC      .+6          ;IF NO ERROR, PROCEED
7511 033614 102002                ESCAPE  SUB          ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
7512 033616                ESCAPE  SUB          ;           AND EXIT THIS TEST          TRAP  C$ESCAPE
7513 033616 104410                ESCAPE  TST          ;           AND EXIT THIS TEST          .WORD L10075-.
7514 033620 002234                BCC      6$          ;NO, GOOD.
7515 033622 103026                BCC      6$          ;YES, REPORT IT'S NOT BEING CLEARED @ INIT.
7516 033622 103026                BCC      6$          ;GET T1CH FOR ERROR MESSAGE
7517 033624 004537 004064        JSR      R5,READ
7518 033630 120005                T1CH
7519 033632 002452                TMP5
7520 033634 103003          BCC      .+10          ;IF NO ERROR, PROCEED
7521 033636                ERROR          ;ELSE, REPORT IT
7522 033636 104460                ESCAPE  TST          ;           AND EXIT THIS TEST          TRAP  C$ERROR
7523 033640                ESCAPE  TST          ;           AND EXIT THIS TEST          TRAP  C$ESCAPE
7524 033640 104410                ESCAPE  TST          ;           AND EXIT THIS TEST          .WORD L10073-.
7525 033642 002214                JSR      R5,READ          ;GET T1LH FOR ERROR MESSAGE
7526 033644 004537 004064        T1LH
7527 033650 120007                TMP7
7528 033652 002456                BCC      .+10          ;IF NO ERROR, PROCEED
7529 033654 103003          ERROR          ;ELSE, REPORT IT
7530 033656                ESCAPE  TST          ;           AND EXIT THIS TEST          TRAP  C$ERROR
7531 033656 104460                ESCAPE  TST          ;           AND EXIT THIS TEST          TRAP  C$ESCAPE
7532 033660                ESCAPE  TST          ;           AND EXIT THIS TEST          .WORD L10073-.
7533 033660 104410                GEDF     EM50B,ERR50    ;REPORT 'T1' NOT CLEARED @ INIT.
7534 033662 002174                GEDF     EM50B,ERR50    ;           'DEVICE FATAL' ERROR # 67
7535 033664                GEDF     EM50B,ERR50    ;           'DEVICE FATAL' ERROR # 67
7536 033664                GEDF     EM50B,ERR50    ;           'DEVICE FATAL' ERROR # 67
7537 033664 104455                TRAP    C$ERDF
7538 033666 000103                .WORD  67
7539 033670 016135                .WORD  EM50B
7540 033672 010762                .WORD  ERR50
7541 033674                ESCAPE  SUB          ;AND EXIT SUBTEST
7542 033674 104410                TRAP    C$ESCAPE
7543 033676 002156                .WORD  L10075-.
7544
7545
7546 -----
7547 033700 004737 003762 6$: JSR      PC,MSTCLR      ;INIT DMV & ENTER M-LOOP AGAIN
7548 033704 112737 000377 002445 MOV     #377,TMP2+1    ;INITIAL VALUE FOR DDRB
7549 033712 004537 004310        JSR      R5,WRITE      ;LOAD IT
7550 033716 120002                DDRB
7551 033720 002445                TMP2+1
7552 033722 103003          BCC      .+10          ;IF NO ERROR, PROCEED
7553 033724                ERROR          ;ELSE, REPORT IT
7554 033724 104460                ESCAPE  TST          ;           AND EXIT THIS TEST          TRAP  C$ERROR
7555 033726                ESCAPE  TST          ;           AND EXIT THIS TEST          TRAP  C$ESCAPE
7556 033726 104410                ESCAPE  TST          ;           AND EXIT THIS TEST          .WORD L10073-.
7557 033730 002126                JSR      R5,INITT1     ;RE-INITIALIZE THE TIMER
7558 033732 004537 004660        0          ;           FOR MAXIMUM TIMEOUT
7559 033736 000000

```



CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

7560 033740 000200          BIT7          ;          MODE 2 & CLEARED 'T1' INT. FLAG
7561 033742 103003          BCC          .+10      ;IF NO ERROR, PROCEED
7562 033744          ERROR          ;ELSE, REPORT IT
7563 033744 104460          ESCAPE TST          ;          AND EXIT THIS TEST          TRAP      C$ERROR
7564 033746          .WORD          ;
7565 033746 104410          .WORD          ;
7566 033750 002106          JSR          R5,READ      ;GET ACR FOR FUTURE ERROR MESSAGES          TRAP      C$ESCAPE
7567 033752 004537 004064          ACR          ;
7568 033756 120013          TMPB          ;
7569 033760 002466          BCC          .+10      ;IF NO ERROR, PROCEED
7570 033762 103003          ERROR          ;ELSE, REPORT IT
7571 033764          ESCAPE TST          ;          AND EXIT THIS TEST          TRAP      C$ERROR
7572 033764 104460          .WORD          ;
7573 033766          .WORD          ;
7574 033766 104410          JSR          R5,LODT1C    ;LOAD TIMER # 1
7575 033770 002066          .BYTE        252
7576 033772 004537 036060          .BYTE        252
7577 033776          .BYTE        252
7578 033777          .BYTE        252
7579
7580
7581
7582 034000 004737 036276          JSR          PC,GETPB7    ;GET 'PB7'. IS IT CLEARED?
7583 034004 102002          BVC          .+6        ;IF NO ERROR, PROCEED
7584 034006          ESCAPE SUB          ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT          TRAP      C$ESCAPE
7585 034006 104410          .WORD          ;
7586 034010 002044          .WORD          ;
7587 034012 103066          BCC          9$         ;IT IS, GOOD
7588 034014 004537 004064          JSR          R5,READ      ;GET IFR FOR ERROR MESSAGE
7589 034020 120015          IFR          ;
7590 034022 002472          TMPD          ;
7591 034024 103003          BCC          .+10      ;IF NO ERROR, PROCEED
7592 034026          ERROR          ;ELSE, REPORT IT
7593 034026 104460          ESCAPE TST          ;          AND EXIT THIS TEST          TRAP      C$ERROR
7594 034030          .WORD          ;
7595 034030 104410          .WORD          ;
7596 034032 002024          .WORD          ;
7597 034034 004537 004064          JSR          R5,READ      ;GET T1CL FOR ERROR MESSAGE
7598 034040 120004          T1CL          ;
7599 034042 002450          TMP4          ;
7600 034044 103003          BCC          .+10      ;IF NO ERROR, PROCEED
7601 034046          ERROR          ;ELSE, REPORT IT
7602 034046 104460          ESCAPE TST          ;          AND EXIT THIS TEST          TRAP      C$ERROR
7603 034050          .WORD          ;
7604 034050 104410          .WORD          ;
7605 034052 002004          .WORD          ;
7606 034054 004537 004064          JSR          R5,READ      ;GET T1CH FOR ERROR MESSAGE
7607 034060 120005          T1CH          ;
7608 034062 002452          TMP5          ;
7609 034064 103003          BCC          .+10      ;IF NO ERROR, PROCEED
7610 034066          ERROR          ;ELSE, REPORT IT
7611 034066 104460          ESCAPE TST          ;          AND EXIT THIS TEST          TRAP      C$ERROR
7612 034070          .WORD          ;
7613 034070 104410          .WORD          ;
7614 034072 001764          .WORD          ;
7615 034074 004537 004064          JSR          R5,READ      ;GET T1LL FOR ERROR MESSAGE

```

CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

7616 034100 120006      T1LL
7617 034102 002454      TMP6
7618 034104 103003      BCC      .+10      ;IF NO ERROR, PROCEED
7619 034106           ERROR      ;ELSE, REPORT IT
7620 034106 104460           ESCAPE TST      ;           AND EXIT THIS TEST      TRAP      CSERROR
7621 034110           ;
7622 034110 104410           ;           AND EXIT THIS TEST      TRAP      C$ESCAPE
7623 034112 001744           ;           AND EXIT THIS TEST      .WORD    L10073-.
7624 034114 004537 004064      JSR      R5,READ      ;GET T1LH FOR ERROR MESSAGE
7625 034120 120007           T1LH
7626 034122 002456           TMP7
7627 034124 103003      BCC      .+10      ;IF NO ERROR, PROCEED
7628 034126           ERROR      ;ELSE, REPORT IT
7629 034126 104460           ESCAPE TST      ;           AND EXIT THIS TEST      TRAP      CSERROR
7630 034130           ;
7631 034130 104410           ;           AND EXIT THIS TEST      TRAP      C$ESCAPE
7632 034132 001724           ;           AND EXIT THIS TEST      .WORD    L10073-.
7633 034134           GEDF      EM50V,ERR50      ;NO, STILL(?) SET!
7634           ;           'DEVICE FATAL' ERROR # 68
7635 034134 104455           ;           'DEVICE FATAL' ERROR # 68      TRAP      CSERDF
7636 034136 000104           ;           'DEVICE FATAL' ERROR # 68      .WORD    68
7637 034140 017240           ;           'DEVICE FATAL' ERROR # 68      .WORD    EM50V
7638 034142 010762           ;           'DEVICE FATAL' ERROR # 68      .WORD    ERR50
7639 034144 004737 036150      JSR      PC,KICKT1      ;BECAUSE THE ERROR MESSAGE TAKES SO LONG TO
7640 034150 103003      BCC      .+10      ;IF NO ERROR, PROCEED
7641 034152           ERROR      ;ELSE, REPORT IT
7642 034152 104460           ESCAPE TST      ;           AND EXIT THIS TEST      TRAP      CSERROR
7643 034154           ;
7644 034154 104410           ;           AND EXIT THIS TEST      TRAP      C$ESCAPE
7645 034156 001700           ;           AND EXIT THIS TEST      .WORD    L10073-.
7646 034160 004737 005032      JSR      PC,STALL      ; PROCESS & PRINT, RE-START THE TIMER AND THEN
7647 034164 004737 005032      JSR      PC,STALL      ; DELAY FOR A LITTLE WHILE SO IT CAN DECREMENT
7648 034170 004537 004064      JSR      R5,READ      ;READ THE LOW COUNTER
7649 034174 120004           T1CL
7650 034176 002450           TMP4
7651 034200 103003      BCC      .+10      ;IF NO ERROR, PROCEED
7652 034202           ERROR      ;ELSE, REPORT IT
7653 034202 104460           ESCAPE TST      ;           AND EXIT THIS TEST      TRAP      CSERROR
7654 034204           ;
7655 034204 104410           ;           AND EXIT THIS TEST      TRAP      C$ESCAPE
7656 034206 001650           ;           AND EXIT THIS TEST      .WORD    L10073-.
7657 034210 123737 002450 033776      CMPB    TMP4,7$      ;MAKE SURE THE COUNTER IS DECREMENTING
7658 034216 001012      BNE      12$           ;IT IS, NOW SEE IF THE HIGH COUNTER IS TOO
7659 034220           GEDF      EM50D,ERR50      ;IT WASN'T -- REPORT THE ERROR
7660           ;           'DEVICE FATAL' ERROR # 69
7661 034220 104455           ;           'DEVICE FATAL' ERROR # 69      TRAP      CSERDF
7662 034222 000105           ;           'DEVICE FATAL' ERROR # 69      .WORD    69
7663 034224 016251           ;           'DEVICE FATAL' ERROR # 69      .WORD    EM50D
7664 034226 010762           ;           'DEVICE FATAL' ERROR # 69      .WORD    ERR50
7665 034230 004737 036150      JSR      PC,KICKT1      ;RESTART TIMER AGAIN IF ERROR MESSAGE PRINTED
7666 034234 103003      BCC      12$           ;IF NO ERROR, PROCEED
7667 034236           ERROR      ;ELSE, REPORT IT
7668 034236 104460           ESCAPE TST      ;           AND EXIT THIS TEST      TRAP      CSERROR
7669 034240           ;
7670 034240 104410           ;           AND EXIT THIS TEST      TRAP      C$ESCAPE
7671 034242 001614           ;           AND EXIT THIS TEST      .WORD    L10073-.

```



CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

7672 034244 012703 000100
7673 034250 004537 004064
7674 034254 120005
7675 034256 002452
7676 034260 103003
7677 034262
7678 034262 104460
7679 034264
7680 034264 104410
7681 034266 001570
7682 034270 123737 002452 033777
7683 034276 001027
7684 034300 077315
7685 034302 004537 004064
7686 034306 120006
7687 034310 002454
7688 034312 103003
7689 034314
7690 034314 104460
7691 034316
7692 034316 104410
7693 034320 001536
7694 034322 004537 004064
7695 034326 120007
7696 034330 002456
7697 034332 103003
7698 034334
7699 034334 104460
7700 034336
7701 034336 104410
7702 034340 001516
7703 034342
7704
7705 034342 104455
7706 034344 000106
7707 034346 016305
7708 034350 010762
7709 034352
7710 034352 104410
7711 034354 001500
7712
7713
7714 034356 112737 000377 002445
7715 034364 004537 004310
7716 034370 120002
7717 034372 002445
7718 034374 103003
7719 034376
7720 034376 104460
7721 034400
7722 034400 104410
7723 034402 001454
7724 034404 004537 036060
7725 034410 001
7726 034411 000
7727
12$: MOV #100,R3 ;INIT. TIMEOUT VALUE
13$: JSR R5,READ ;READ THE HIGH COUNTER
T1CH
TMP5
BCC .+10 ;IF NO ERROR, PROCEED
ERROR ;ELSE, REPORT IT
ESCAPE TST ; AND EXIT THIS TEST TRAP C$ERROR
; TRAP C$ESCAPE
; .WORD L10073-.
CMPB TMP5,8$ ;DID IT CHANGE FROM THE LOADED VALUE?
BNE 17$ ;YES, PROCEED WITH TESTING
SOB R3,13$ ;NO, IF NO TIMEOUT, TRY AGAIN
JSR R5,READ ;GET T1LL FOR ERROR MESSAGE
T1LL
TMP6
BCC .+10 ;IF NO ERROR, PROCEED
ERROR ;ELSE, REPORT IT
ESCAPE TST ; AND EXIT THIS TEST TRAP C$ERROR
; TRAP C$ESCAPE
; .WORD L10073-.
JSR R5,READ ;GET T1LH FOR ERROR MESSAGE
T1LH
TMP7
BCC .+10 ;IF NO ERROR, PROCEED
ERROR ;ELSE, REPORT IT
ESCAPE TST ; AND EXIT THIS TEST TRAP C$ERROR
; TRAP C$ESCAPE
; .WORD L10073-.
GEDF EM50E,ERR50 ;ELSE, REPORT THAT HIGH COUNTER ISN'T RUNNING
; 'DEVICE FATAL' ERROR # 70
TRAP C$ERDF
; .WORD 70
; .WORD EM50E
; .WORD ERR50
ESCAPE SUB ;IN THAT CASE, WE CAN'T PROCEED WITH TESTING EITHER
TRAP C$ESCAPE
; .WORD L10075-.
-----
17$: MOVB #377,TMP2+1 ;SETUP DDRB FOR DESIRED DIRECTION OF ORB
JSR R5,WRITE
DDR
TMP2+1
BCC .+10 ;IF NO ERROR, PROCEED
ERROR ;ELSE, REPORT IT
ESCAPE TST ; AND EXIT THIS TEST TRAP C$ERROR
; TRAP C$ESCAPE
; .WORD L10073-.
JSR R5,LODT1C ;RE-LOAD TIMER # 1 WITH A VALUE WHICH WILL
18$: .BYTE 1 ; CAUSE AN ALMOST IMMEDIATE TIMEOUT
19$: .BYTE 0 ; (ADDRESS OF HIGH BYTE FOR T1C-H (ADDR 05))
-----

```

CVDMAA.P11

12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

7728
7729 034412 004737 036112      JSR    PC,GETT1      ;WAS 'T1' SET BY THE ABOVE OPERATION?
7730 034416 102002      BVC    .+6           ;IF NO ERROR, PROCEED
7731 034420      ESCAPE SUB           ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
7732 034420 104410      TRAP   C$ESCAPE     ;
7733 034422 001432      .WORD  L10075-.
7734 034424 103446      BCS    20$          ;YES, OK -- CONTINUE ERROR CHECKING
7735 034426 004537 004064      JSR    R5,READ      ;GET T1CL FOR ERROR MESSAGE
7736 034432 120004      T1CL
7737 034434 002450      TMP4
7738 034436 103003      BCC    .+10         ;IF NO ERROR, PROCEED
7739 034440      ERROR              ;ELSE, REPORT IT
7740 034440 104460      TRAP   C$ERROR
7741 034442      ESCAPE TST         ; AND EXIT THIS TEST
7742 034442 104410      TRAP   C$ESCAPE     ;
7743 034444 001412      .WORD  L10073-.
7744 034446 004537 004064      JSR    R5,READ      ;GET T1CH FOR ERROR MESSAGE
7745 034452 120005      T1CH
7746 034454 002452      TMP5
7747 034456 103003      BCC    .+10         ;IF NO ERROR, PROCEED
7748 034460      ERROR              ;ELSE, REPORT IT
7749 034460 104460      TRAP   C$ERROR
7750 034462      ESCAPE TST         ; AND EXIT THIS TEST
7751 034462 104410      TRAP   C$ESCAPE     ;
7752 034464 001372      .WORD  L10073-.
7753 034466 004537 004064      JSR    R5,READ      ;GET T1LL FOR ERROR MESSAGE
7754 034472 120006      T1LL
7755 034474 002454      TMP6
7756 034476 103003      BCC    .+10         ;IF NO ERROR, PROCEED
7757 034500      ERROR              ;ELSE, REPORT IT
7758 034500 104460      TRAP   C$ERROR
7759 034502      ESCAPE TST         ; AND EXIT THIS TEST
7760 034502 104410      TRAP   C$ESCAPE     ;
7761 034504 001352      .WORD  L10073-.
7762 034506 004537 004064      JSR    R5,READ      ;GET T1LH FOR ERROR MESSAGE
7763 034512 120007      T1LH
7764 034514 002456      TMP7
7765 034516 103003      BCC    .+10         ;IF NO ERROR, PROCEED
7766 034520      ERROR              ;ELSE, REPORT IT
7767 034520 104460      TRAP   C$ERROR
7768 034522      ESCAPE TST         ; AND EXIT THIS TEST
7769 034522 104410      TRAP   C$ESCAPE     ;
7770 034524 001332      .WORD  L10073-.
7771 034526      GEDF  EM50F,ERR50 ;NO, BAD NEWS! REPORT THE FAILURE
7772      ; 'DEVICE FATAL' ERROR # 71
7773 034526 104455      TRAP   C$ERDF      ;
7774 034530 000107      .WORD  71
7775 034532 016341      .WORD  EM50F
7776 034534 010762      .WORD  ERR50
7777 034536      ESCAPE SUB         ; AND GET OUT OF SUBTEST
7778 034536 104410      TRAP   C$ESCAPE     ;
7779 034540 001314      .WORD  L10075-.
7780 034542 004737 036276      20$: JSR    PC,GETPB7   ;GET 'PB7'. IS IT SET?
7781 034546 102002      BVC    .+6           ;IF NO ERROR, PROCEED
7782 034550      ESCAPE SUB         ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
7783 034550 104410      TRAP   C$ESCAPE

```



CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

7784	034552	001302							.WORD	L10075-.
7785	034554	103445		BCS	41\$		:YES, CONTINUE CHECKING 'T1'			
7786	034556	004537	004064	JSR	R5,READ		:GET T1CL FOR ERROR MESSAGE			
7787	034562	120004		T1CL						
7788	034564	002450		TMP4						
7789	034566	103003		BCC	.+10		:IF NO ERROR, PROCEED			
7790	034570			ERROR			:ELSE, REPORT IT			
7791	034570	104460		ESCAPE	TST		: AND EXIT THIS TEST	TRAP		C\$ERROR
7792	034572									
7793	034572	104410						TRAP		C\$ESCAPE
7794	034574	001262						.WORD		L10073-.
7795	034576	004537	004064	JSR	R5,READ		:GET T1CH FOR ERROR MESSAGE			
7796	034602	120005		T1CH						
7797	034604	002452		TMP5						
7798	034606	103003		BCC	.+10		:IF NO ERROR, PROCEED			
7799	034610			ERROR			:ELSE, REPORT IT			
7800	034610	104460		ESCAPE	TST		: AND EXIT THIS TEST	TRAP		C\$ERROR
7801	034612									
7802	034612	104410						TRAP		C\$ESCAPE
7803	034614	001242						.WORD		L10073-.
7804	034616	004537	004064	JSR	R5,READ		:GET T1LL FOR ERROR MESSAGE			
7805	034622	120006		T1LL						
7806	034624	002454		TMP6						
7807	034626	103003		BCC	.+10		:IF NO ERROR, PROCEED			
7808	034630			ERROR			:ELSE, REPORT IT			
7809	034630	104460		ESCAPE	TST		: AND EXIT THIS TEST	TRAP		C\$ERROR
7810	034632									
7811	034632	104410						TRAP		C\$ESCAPE
7812	034634	001222						.WORD		L10073-.
7813	034636	004537	004064	JSR	R5,READ		:GET T1LH FOR ERROR MESSAGE			
7814	034642	120007		T1LH						
7815	034644	002456		TMP7						
7816	034646	103003		BCC	.+10		:IF NO ERROR, PROCEED			
7817	034650			ERROR			:ELSE, REPORT IT			
7818	034650	104460		ESCAPE	TST		: AND EXIT THIS TEST	TRAP		C\$ERROR
7819	034652									
7820	034652	104410						TRAP		C\$ESCAPE
7821	034654	001202						.WORD		L10073-.
7822	034656			GEDF	EM50S,ERR50		:NO! REPORT THAT PB7 DIDN'T GET SET!			
7823							: 'DEVICE FATAL' ERROR # 72			
7824	034656	104455						TRAP		C\$ERDF
7825	034660	000110						.WORD		72
7826	034662	017122						.WORD		EM50S
7827	034664	010762						.WORD		ERR50
7828	034666	000562		BR	28\$		: & EXIT THIS SECTION OF SUBTEST			
7829	034670	004537	004064	JSR	R5,READ	41\$:	:READ T1C-H (ADDR 05) TO SEE IF IT CLEARS 'T1'			
7830	034674	120005		T1CH			:(THIS VALUE ISN'T CHECKED BECAUSE IT CAN BE			
7831	034676	002452		TMP5			: ALMOST ANYTHING)			
7832	034700	103003		BCC	.+10		:IF NO ERROR, PROCEED			
7833	034702			ERROR			:ELSE, REPORT IT			
7834	034702	104460		ESCAPE	TST		: AND EXIT THIS TEST	TRAP		C\$ERROR
7835	034704									
7836	034704	104410						TRAP		C\$ESCAPE
7837	034706	001150						.WORD		L10073-.
7838	034710	004737	036112	JSR	PC,GETT1		:PUT THE CURRENT 'T1' VALUE INTO THE CARRY BIT			
7839	034714	102002		BVC	11\$		:IF NO ERROR, PROCEED			

CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

7840 034716          ESCAPE SUB          ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
7841 034716 104410          TRAP C$ESCAPE
7842 034720 001134          .WORD L10075-.
7843 034722 103435          11$: BCS 21$          ;IF SET, ALL'S OK
7844          ;IF CLEARED! BAD VIA CHIP!
7845 034724 004537 004064 JSR R5,READ          ;GET T1CL FOR ERROR MESSAGE
7846 034730 120004          T1CL
7847 034732 002450          TMP4
7848 034734 103003          BCC .+10          ;IF NO ERROR, PROCEED
7849 034736          ERROR          ;ELSE, REPORT IT
7850 034736 104460          ESCAPE TST          ; AND EXIT THIS TEST
7851 034740          TRAP C$ERROR
7852 034740 104410          ; AND EXIT THIS TEST
7853 034742 001114          TRAP C$ESCAPE
7854 034744 004537 004064 JSR R5,READ          ;GET T1LL FOR ERROR MESSAGE
7855 034750 120006          T1LL
7856 034752 002454          TMP6
7857 034754 103003          BCC .+10          ;IF NO ERROR, PROCEED
7858 034756          ERROR          ;ELSE, REPORT IT
7859 034756 104460          ESCAPE TST          ; AND EXIT THIS TEST
7860 034760          TRAP C$ERROR
7861 034760 104410          ; AND EXIT THIS TEST
7862 034762 001074          TRAP C$ESCAPE
7863 034764 004537 004064 JSR R5,READ          ;GET T1LH FOR ERROR MESSAGE
7864 034770 120007          T1LH
7865 034772 002456          TMP7
7866 034774 103003          BCC .+10          ;IF NO ERROR, PROCEED
7867 034776          ERROR          ;ELSE, REPORT IT
7868 034776 104460          ESCAPE TST          ; AND EXIT THIS TEST
7869 035000          TRAP C$ERROR
7870 035000 104410          ; AND EXIT THIS TEST
7871 035002 001054          TRAP C$ESCAPE
7872 035004          GEDF EM50G,ERR50 ;REPORT BAD VIA CHIP!
7873          ; 'DEVICE FATAL' ERROR # 73
7874 035004 104455          TRAP C$ERDF
7875 035006 000111          .WORD 73
7876 035010 016406          .WORD EM50G
7877 035012 010762          .WORD ERR50
7878 035014 000507          BR 28$          ;BYPASS THE REST OF THIS SECTION OF TESTING
7879          21$: JSR R5,READ          ;READ T1L-L (ADDR 06)
7880 035016 004537 004064 T1LL
7881 035022 120006          TMP6
7882 035024 002454          BCC .+10          ;THIS SHOULD RETURN A 001
7883 035026 103003          ERROR          ;IF NO ERROR, PROCEED
7884 035030          ;ELSE, REPORT IT
7885 035030 104460          ESCAPE TST          ; AND EXIT THIS TEST
7886 035032          TRAP C$ERROR
7887 035032 104410          ; AND EXIT THIS TEST
7888 035034 001022          TRAP C$ESCAPE
7889 035036 123737 002454 034410 CMPB TMP6,18$          ;CHECK T1L-L (ADDR 06) AGAINST LOADED VALUE
7890 035044 001415          BEQ 23$          ;IF SAME, PROCEED
7891          ;ELSE, REPORT BAD LOAD OF T1L-L (ADDR 06)
7892 035046 004537 004064 JSR R5,READ          ;GET T1LH FOR ERROR MESSAGE
7893 035052 120007          T1LH
7894 035054 002456          TMP7
7895 035056 103003          BCC .+10          ;IF NO ERROR, PROCEED

```



CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

7896 035060          ERROR          ;ELSE, REPORT IT
7897 035060 104460          ESCAPE TST          ;          AND EXIT THIS TEST          TRAP      C$ERROR
7898 035062          GEDF      EM50H,ERR50 ;ELSE, REPORT BAD LOAD OF T1L-L (ADDR 06) TRAP      C$ESCAPE
7899 035062 104410          ;          'DEVICE FATAL' ERROR # 74          .WORD    L10073-.
7900 035064 000772          ;          TRAP      C$ERDF
7901 035066          ;          ;          .WORD    74
7902          ;          ;          .WORD    EM50H
7903 035066 104455          ;          ;          .WORD    ERR50
7904 035070 000112          BR      28$          ;BYPASS THE REST OF THIS SECTION OF TESTING
7905 035072 016450          JSR      PC,GETT1          ;IS 'T1' STILL SET?
7906 035074 010762          BVC      .+6          ;IF NO ERROR, PROCEED
7907 035076 000456          ESCAPE  SUB          ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
7908          ;          TRAP      C$ESCAPE
7909 035100 004737 036112 23$: JSR      PC,GETT1          ;          .WORD    L10075-.
7910 035104 102002          BVC      .+6
7911 035106          ESCAPE  SUB
7912 035106 104410          BCS      24$          ;YES, ALL'S OK
7913 035110 000744          ;NO! BAD VIA CHIP!
7914 035112 103415          BCS      24$          ;GET T1LH FOR ERROR MESSAGE
7915          ;          ;
7916 035114 004537 004064 JSR      R5,READ
7917 035120 120007          T1LH
7918 035122 002456          TMP7
7919 035124 103003          BCC      .+10          ;IF NO ERROR, PROCEED
7920 035126          ERROR          ;ELSE, REPORT IT
7921 035126 104460          ESCAPE  TST          ;          AND EXIT THIS TEST          TRAP      C$ERROR
7922 035130          ;          ;          TRAP      C$ESCAPE
7923 035130 104410          ;          ;          .WORD    L10073-.
7924 035132 000724          GEDF      EM50I,ERR50 ;REPORT BAD VIA CHIP!
7925 035134          ;          ;          'DEVICE FATAL' ERROR # 75
7926          ;          ;          TRAP      C$ERDF
7927 035134 104455          ;          ;          .WORD    75
7928 035136 000113          ;          ;          .WORD    EM50I
7929 035140 016536          ;          ;          .WORD    ERR50
7930 035142 010762          BR      28$          ;BYPASS THE REST OF THIS SECTION OF TESTING
7931 035144 000433          ;READ T1L-H (ADDR 07)
7932          ;THIS SHOULD RETURN A 000
7933 035146 004537 004064 24$: JSR      R5,READ
7934 035152 120007          T1LH
7935 035154 002456          TMP7
7936 035156 103003          BCC      .+10          ;IF NO ERROR, PROCEED
7937 035160          ERROR          ;ELSE, REPORT IT
7938 035160 104460          ESCAPE  TST          ;          AND EXIT THIS TEST          TRAP      C$ERROR
7939 035162          ;          ;          TRAP      C$ESCAPE
7940 035162 104410          ;          ;          .WORD    L10073-.
7941 035164 000672          CMPB     TMP7,19$          ;CHECK T1L-H (ADDR 07) AGAINST LOADED VALUE
7942 035166 123737 002456 034411 BEQ      26$
7943 035174 001405          GEDF     EM50J,ERR50 ;IF SAME, PROCEED
7944          ;ELSE, REPORT BAD LOAD OF T1L-H (ADDR 07)
7945          ;          ;          'DEVICE FATAL' ERROR # 76
7946 035176 104455          ;          ;          TRAP      C$ERDF
7947 035200 000114          ;          ;          .WORD    76
7948 035202 016600          ;          ;          .WORD    EM50J
7949 035204 010762          ;          ;          .WORD    ERR50
7950 035206 000412          BR      28$          ;BYPASS THE REST OF THIS SECTION OF TESTING
7951

```

CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

7952 035210 004737 036112
7953 035214 102002
7954 035216
7955 035216 104410
7956 035220 000634
7957 035222 103404
7958 035224
7959
7960 035224 104455
7961 035226 000115
7962 035230 016666
7963 035232 010762
7964
7965
7966
7967 035234 004537 004064
7968 035240 120004
7969 035242 002450
7970 035244 103003
7971 035246
7972 035246 104460
7973 035250
7974 035250 104410
7975 035252 000604
7976 035254 004737 036112
7977 035260 102002
7978 035262
7979 035262 104410
7980 035264 000570
7981 035266 103004
7982 035270
7983
7984 035270 104455
7985 035272 000116
7986 035274 016203
7987 035276 010762
7988
7989
7990
7991 035300 105037 002445
7992 035304 004537 004310
7993 035310 120002
7994 035312 002445
7995 035314 103003
7996 035316
7997 035316 104460
7998 035320
7999 035320 104410
8000 035322 000534
8001 035324 004537 004310
8002 035330 120005
8003 035332 002453
8004 035334 103003
8005 035336
8006 035336 104460
8007 035340

```

```

26$: JSR PC,GETT1 ;IS 'T1' STILL SET?
      BVC .+6 ;IF NO ERROR, PROCEED
      ESCAPE SUB ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
                                         TRAP C$ESCAPE
                                         .WORD L10075-.

      BCS 28$ ;YES, ALL'S OK
      GEDF EM50K,ERR50 ;NO! BAD VIA CHIP!
      ; 'DEVICE FATAL' ERROR # 77
                                         TRAP C$ERDF
                                         .WORD 77
                                         .WORD EM50K
                                         .WORD ERR50

```

---

```

28$: JSR R5,READ ;READ T1C-L (ADDR 04) TO CLEAR 'T1'
      T1CL ;(THIS VALUE ISN'T CHECKED BECAUSE IT CAN BE
      TMP4 ;ALMOST ANYTHING)
      BCC .+10 ;IF NO ERROR, PROCEED
      ERROR ;ELSE, REPORT IT
                                         TRAP C$ERROR
      ESCAPE TST ; AND EXIT THIS TEST
                                         TRAP C$ESCAPE
                                         .WORD L10073-.

      JSR PC,GETT1 ;IS 'T1' CLEARED NOW
      BVC 16$ ;IF NO ERROR, PROCEED
      ESCAPE SUB ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
                                         TRAP C$ESCAPE
                                         .WORD L10075-.

16$: BCC 29$ ;YES, ALL'S OK
      GEDF EM50C,ERR50 ;NO! BAD VIA CHIP!
      ; 'DEVICE FATAL' ERROR # 78
                                         TRAP C$ERDF
                                         .WORD 78
                                         .WORD EM50C
                                         .WORD ERR50

```

---

```

29$: CLRB TMP2+1 ;CHANGE THE DIRECTION OF ORB -- IT SHOULDN'T
      JSR R5,WRITE ; HAVE ANY EFFECT ON 'PB7'
      DDRB
      TMP2+1
      BCC .+10 ;IF NO ERROR, PROCEED
      ERROR ;ELSE, REPORT IT
                                         TRAP C$ERROR
      ESCAPE TST ; AND EXIT THIS TEST
                                         TRAP C$ESCAPE
                                         .WORD L10073-.

      JSR R5,WRITE ;RE-WRITE INTO T1C-H (ADDR 05) TO SET T1 AGAIN
      T1CH
      TMP5+1
      BCC .+10 ;IF NO ERROR, PROCEED
      ERROR ;ELSE, REPORT IT
                                         TRAP C$ERROR
      ESCAPE TST ; AND EXIT THIS TEST

```



CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

8008 035340 104410
8009 035342 000514
8010 035344 004737 036112
8011 035350 102002
8012 035352
8013 035352 104410
8014 035354 000500
8015 035356 103426
8016 035360 004537 004064
8017 035364 120005
8018 035366 002452
8019 035370 103003
8020 035372
8021 035372 104460
8022 035374
8023 035374 104410
8024 035376 000460
8025 035400 004537 004064
8026 035404 120007
8027 035406 002456
8028 035410 103003
8029 035412
8030 035412 104460
8031 035414
8032 035414 104410
8033 035416 000440
8034 035420
8035
8036 035420 104455
8037 035422 000117
8038 035424 016730
8039 035426 010762
8040 035430
8041 035430 104410
8042 035432 000422
8043
8044
8045
-----
8046 035434 004737 036276
8047 035440 102002
8048 035442
8049 035442 104410
8050 035444 000410
8051 035446 103404
8052 035450
8053
8054 035450 104455
8055 035452 000120
8056 035454 017174
8057 035456 010762
8058 035460 112737 000125 002455 44$:
8059 035466 004537 004310
8060 035472 120006
8061 035474 002455
8062 035476 103003
8063 035500

      JSR      PC,GETT1      ;IS 'T1' SET AGAIN
      BVC     .+6            ;IF NO ERROR, PROCEED
      ESCAPE  SUB           ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
                                TRAP      C$ESCAPE
                                .WORD    L10073-.

      BCS     32$           ;YES, ALL'S WELL (AGAIN?)
      JSR     R5,READ       ;GET T1CH FOR ERROR MESSAGE
      T1CH
      TMP5
      BCC     .+10         ;IF NO ERROR, PROCEED
      ERROR   ;ELSE, REPORT IT
                                TRAP      C$ERROR
                                .WORD    L10073-.

      ESCAPE  TST           ;      AND EXIT THIS TEST
                                TRAP      C$ESCAPE
                                .WORD    L10073-.

      JSR     R5,READ       ;GET T1LH FOR ERROR MESSAGE
      T1LH
      TMP7
      BCC     .+10         ;IF NO ERROR, PROCEED
      ERROR   ;ELSE, REPORT IT
                                TRAP      C$ERROR
                                .WORD    L10073-.

      ESCAPE  TST           ;      AND EXIT THIS TEST
                                TRAP      C$ESCAPE
                                .WORD    L10073-.

      GEDF    EM50L,ERR50   ;NO! SOMETHING WENT WRONG! REPORT IT
                                ;      'DEVICE FATAL' ERROR # 79
                                TRAP      C$ERDF
                                .WORD    79
                                .WORD    EM50L
                                .WORD    ERR50

      ESCAPE  SUB           ;      AND EXIT FROM THIS SUBTEST
                                TRAP      C$ESCAPE
                                .WORD    L10075-.

-----
32$: JSR      PC,GETPB7     ;GET 'PB7'. IS IT SET?
      BVC     .+6            ;IF NO ERROR, PROCEED
      ESCAPE  SUB           ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
                                TRAP      C$ESCAPE
                                .WORD    L10075-.

      BCS     44$           ;YES, GOOD.
      GEDF    EM50U,ERR50   ;NO, BAD! REPORT IT: NL? SET AFTER TIMEOUT
                                ;      'DEVICE FATAL' ERROR # 80
                                TRAP      C$ERDF
                                .WORD    80
                                .WORD    EM50U
                                .WORD    ERR50

      MOVB    #125,TMP6+1   ;USING A DIFFERENT VALUE -- 55 HEX.,
      JSR     R5,WRITE       ;RE-LOAD T1L-L (ADDR 06)
      T1LL
      TMP6+1
      BCC     .+10         ;IF NO ERROR, PROCEED
      ERROR   ;ELSE, REPORT IT
  
```

CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

8064 035500 104460
8065 035502          ESCAPE TST          ;          AND EXIT THIS TEST          TRAP  CSERROR
8066 035502 104410
8067 035504 000352          .WORD          CS$ESCAPE
8068 035506 004737 036112      JSR    PC,GETT1          ;IS 'T1' STILL SET?
8069 035512 102002          BVC    .+6              ;IF NO ERROR, PROCEED
8070 035514          ESCAPE SUB              ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
8071 035514 104410          TRAP  CS$ESCAPE
8072 035516 000336          .WORD          L10073-.
8073 035520 103416
8074 035522 004537 004064      BCS    33$
8075 035526 120006          JSR    R5,READ          ;YES, ALL'S STILL OK
8076 035530 002454          TILL  ;GET TILL FOR ERROR MESSAGE
8077 035532 103003          TMP6
8078 035534          BCC    .+10            ;IF NO ERROR, PROCEED
8079 035534 104460          ERROR          ;ELSE, REPORT IT
8080 035536          ESCAPE TST          ;          AND EXIT THIS TEST          TRAP  CSERROR
8081 035536 104410          TRAP  CS$ESCAPE
8082 035540 000316          .WORD          L10073-.
8083 035542          GEDF   EM50M,ERR50      ;NO! SOMETHING WENT WRONG! REPORT IT
8084          ;          'DEVICE FATAL' ERROR # 81
8085 035542 104455          TRAP  CS$ERDF
8086 035544 000121          .WORD          81
8087 035546 017012          .WORD          EM50M
8088 035550 010762          .WORD          ERR50
8089 035552          ESCAPE SUB          ;          AND EXIT FROM THIS SUBTEST
8090 035552 104410          TRAP  CS$ESCAPE
8091 035554 000300          .WORD          L10073-.
8092
8093
8094
-----
8095 035556 112737 000125 002453 33$:  MOVB   #125,TMP5+1      ;AND USING THE SAME VALUE AGAIN (55 HEX),
8096 035564 004537 004310      JSR    R5,WRITE          ;NOW LOAD TIC-H (ADDR 05)
8097 035570 120005          T1CH
8098 035572 002453          TMP5+1
8099 035574 103003          BCC    .+10            ;IF NO ERROR, PROCEED
8100 035576          ERROR          ;ELSE, REPORT IT
8101 035576 104460          TRAP  CSERROR
8102 035600          ESCAPE TST          ;          AND EXIT THIS TEST          TRAP  CS$ESCAPE
8103 035600 104410          .WORD          L10073-.
8104 035602 000254
8105 035604 004737 036112      JSR    PC,GETT1          ;'T1' SHOULD NOW BE CLEARED
8106 035610 102002          BVC    .+6              ;IF NO ERROR, PROCEED
8107 035612          ESCAPE SUB              ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
8108 035612 104410          TRAP  CS$ESCAPE
8109 035614 000240          .WORD          L10075-.
8110 035616 103024          BCC    34$
8111 035620 004537 004064      JSR    R5,READ          ;IT WAS, ALL'S WELL THAT END'S WELL (I THINK!?)
8112 035624 120006          TILL  ;GET TILL FOR ERROR MESSAGE
8113 035626 002454          TMP6
8114 035630 103003          BCC    .+10            ;IF NO ERROR, PROCEED
8115 035632          ERROR          ;ELSE, REPORT IT
8116 035632 104460          TRAP  CSERROR
8117 035634          ESCAPE TST          ;          AND EXIT THIS TEST          TRAP  CS$ESCAPE
8118 035634 104410          .WORD          L10073-.
8119 035636 000220

```



CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

8120 035640 004537 004064      JSR      R5,READ      ;GET T1LH FOR ERROR MESSAGE
8121 035644 120007              T1LH
8122 035646 002456              TMP7
8123 035650 103003      BCC      .+10        ;IF NO ERROR, PROCEED
8124 035652 104460      ERROR      ;ELSE, REPORT IT
8125 035652 104460              ESCAPE  TST          ;          AND EXIT THIS TEST          TRAP      C$ERROR
8126 035654 104410              ESCAPE  TST          ;          AND EXIT THIS TEST          TRAP      C$ESCAPE
8127 035654 104410              ESCAPE  TST          ;          AND EXIT THIS TEST          .WORD    L10073-.
8128 035656 000200              GEDF    EM50N,ERR50  ;IT WASN'T! SOMETHING WENT WRONG! REPORT IT
8129 035660              GEDF    EM50N,ERR50  ;          'DEVICE FATAL' ERROR # 82
8130 035660 104455              GEDF    EM50N,ERR50  ;          'DEVICE FATAL' ERROR # 82          TRAP      C$ERDF
8131 035660 104455              GEDF    EM50N,ERR50  ;          'DEVICE FATAL' ERROR # 82          .WORD    82
8132 035662 000122              GEDF    EM50N,ERR50  ;          'DEVICE FATAL' ERROR # 82          .WORD    EM50N
8133 035664 017054              GEDF    EM50N,ERR50  ;          'DEVICE FATAL' ERROR # 82          .WORD    ERR50
8134 035666 010762              GEDF    EM50N,ERR50  ;          'DEVICE FATAL' ERROR # 82
8135 035666 010762              GEDF    EM50N,ERR50  ;          'DEVICE FATAL' ERROR # 82
8136 035670 004537 004310      34$: JSR      R5,WRITE  ;RE-LOAD T1C-H (ADDR 5) TO START IT AGAIN
8137 035674 120005              T1CH
8138 035676 002453              TMP5+1
8139 035700 103003      BCC      .+10        ;IF NO ERROR, PROCEED
8140 035702 103003      ERROR      ;ELSE, REPORT IT
8141 035702 104460              ESCAPE  TST          ;          AND EXIT THIS TEST          TRAP      C$ERROR
8142 035704 104460              ESCAPE  TST          ;          AND EXIT THIS TEST          TRAP      C$ESCAPE
8143 035704 104410              ESCAPE  TST          ;          AND EXIT THIS TEST          .WORD    L10073-.
8144 035706 000150              ESCAPE  TST          ;          AND EXIT THIS TEST          TRAP      C$ESCAPE
8145 035710 004737 036276      JSR      PC,GETPB7   ;GET 'PB7'. IS IT CLEARED?
8146 035714 102002              BVC     .+6          ;IF NO ERROR, PROCEED
8147 035716 104410              ESCAPE  SUB          ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT          TRAP      C$ESCAPE
8148 035716 104410              ESCAPE  SUB          ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT          .WORD    L10075-.
8149 035720 000134              ESCAPE  SUB          ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
8150 035722 103054              BCC     48$         ;YES, GOOD.
8151 035724 004537 004064      JSR      R5,READ      ;GET IFR FOR ERROR MESSAGE
8152 035730 120015              IFR
8153 035732 002472              TMPD
8154 035734 103003      BCC      .+10        ;IF NO ERROR, PROCEED
8155 035736 103003      ERROR      ;ELSE, REPORT IT
8156 035736 104460              ESCAPE  TST          ;          AND EXIT THIS TEST          TRAP      C$ERROR
8157 035740 104460              ESCAPE  TST          ;          AND EXIT THIS TEST          TRAP      C$ESCAPE
8158 035740 104410              ESCAPE  TST          ;          AND EXIT THIS TEST          .WORD    L10073-.
8159 035742 000114              ESCAPE  TST          ;          AND EXIT THIS TEST          TRAP      C$ESCAPE
8160 035744 004537 004064      JSR      R5,READ      ;GET T1CL FOR ERROR MESSAGE
8161 035750 120004              T1CL
8162 035752 002450              TMP4
8163 035754 103003      BCC      .+10        ;IF NO ERROR, PROCEED
8164 035756 103003      ERROR      ;ELSE, REPORT IT
8165 035756 104460              ESCAPE  TST          ;          AND EXIT THIS TEST          TRAP      C$ERROR
8166 035760 104460              ESCAPE  TST          ;          AND EXIT THIS TEST          TRAP      C$ESCAPE
8167 035760 104410              ESCAPE  TST          ;          AND EXIT THIS TEST          .WORD    L10073-.
8168 035762 000074              ESCAPE  TST          ;          AND EXIT THIS TEST          TRAP      C$ESCAPE
8169 035764 004537 004064      JSR      R5,READ      ;GET T1CH FOR ERROR MESSAGE
8170 035770 120005              T1CH
8171 035772 002452              TMP5
8172 035774 103003      BCC      .+10        ;IF NO ERROR, PROCEED
8173 035776 103003      ERROR      ;ELSE, REPORT IT
8174 035776 104460              ESCAPE  TST          ;          AND EXIT THIS TEST          TRAP      C$ERROR
8175 036000 104460              ESCAPE  TST          ;          AND EXIT THIS TEST          TRAP      C$ERROR

```

CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

8176 036000 104410
8177 036002 000054
8178 036004 004537 004064
8179 036010 120006
8180 036012 002454
8181 036014 103003
8182 036016
8183 036016 104460
8184 036020
8185 036020 104410
8186 036022 000034
8187 036024 004537 004064
8188 036030 120007
8189 036032 002456
8190 036034 103003
8191 036036
8192 036036 104460
8193 036040
8194 036040 104410
8195 036042 000014
8196 036044
8197
8198 036044 104455
8199 036046 000123
8200 036050 017240
8201 036052 010762
8202 036054
8203 036054
8204 036054 104403
8205 036056
8206 036056
8207 036056 104401
8208
8209
8210
8211
8212
8213
8214
8215
8216
8217
8218
8219
8220
8221 036060 112537 002451
8222 036064 112537 002453
8223 036070 004537 004310
8224 036074 120004
8225 036076 002451
8226 036100 004537 004310
8227 036104 120005
8228 036106 002453
8229 036110 000205
8230
8231

```

```

          JSR      R5,READ          ;GET T1LL FOR ERROR MESSAGE
          T1LL
          TMP6
          BCC     .+10             ;IF NO ERROR, PROCEED
          ERROR    ;ELSE, REPORT IT
          ESCAPE  TST              ; AND EXIT THIS TEST
          JSR      R5,READ          ;GET T1LH FOR ERROR MESSAGE
          T1LH
          TMP7
          BCC     .+10             ;IF NO ERROR, PROCEED
          ERROR    ;ELSE, REPORT IT
          ESCAPE  TST              ; AND EXIT THIS TEST
          GEDF    EM50V,ERR50      ;NO, BAD! RPT. PB7 NOT DRIVEN LOW
          ; 'DEVICE FATAL' ERROR # 83
          TRAP    C$ESCAPE
          .WORD   L10073-
          TRAP    C$ERROR
          TRAP    C$ESCAPE
          .WORD   L10073-
          TRAP    C$ERROR
          TRAP    C$ESCAPE
          .WORD   L10073-
          TRAP    C$ERDF
          .WORD   83
          .WORD   EM50V
          .WORD   ERR50
48$:     ENDSUB
          TRAP    L10075:
          TRAP    C$ESUB
          TRAP    L10073:
          TRAP    C$ETST

```

```

-----
: LODT1C -- LOAD TIMER ONE AT ADDRESSES 04 & 05
: CALLING SEQUENCE:
:
:     JSR      R5,LODT1C
:     .BYTE   <VALUE FOR T1L-L (ADDRESS 04)>
:     .BYTE   <VALUE FOR T1C-H (ADDRESS 05)>
:     <NEXT SEQUENTIAL INSTRUCTION
:
-----
LODT1C: MOVB    (R5)+,TMP4+1      ;SETUP TO LOAD T1CL
        MOVB    (R5)+,TMP5+1      ; AND T1CH
        JSR     R5,WRITE          ;LOAD T1C-L (ADDR 04) WITH PASSED PARAMETER
        T1CL
        TMP4+1
        JSR     R5,WRITE          ;LOAD T1C-H (ADDR 05) WITH PASSED PARAMETER
        T1CH                      ; (THIS WILL ALSO RESET 'T1' & THE COUNTER)
        TMP5+1
        RTS     R5
:
-----

```



CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

```

8232
8233
8234
8235
8236
8237 036112 004537 004064
8238 036116 120015
8239 036120 002472
8240 036122 103003
8241 036124
8242 036124 104460
8243 036126 000262
8244 036130 000207
8245
8246 036132 010046
8247 036134 113700 002472
8248 036140 106100
8249 036142 106100
8250 036144 012600
8251 036146 000207
8252
8253
8254
8255
8256
8257
8258
8259
8260
8261
8262
8263
8264 036150 010346
8265 036152 004537 004064
8266 036156 120007
8267 036160 002456
8268 036162 103443
8269
8270 036164 012777 120005 144170
8271 036172 113777 002456 144166
8272 036200 113737 002456 002453
8273 036206 142777 000100 144144
8274 036214 112777 000002 144134
8275 036222 012703 000074
8276 036226 132777 000200 144122
8277 036234 001016
8278 036236 077305
8279 036240
8280
8281 036240 012737 000001 002236
8282 036246 012737 000124 002240
8283 036254 012737 014500 002242
8284 036262 012737 005426 002244
8285 036270 000261
8286
8287 036272 012603

```

```

: GETT1 -- GET THE 'T1' FLAG FROM THE VIA'S IFR REGISTER AND PUT IT
: INTO THE 'CARRY' BIT
-----
GETT1: JSR R5,READ ;GET VIA'S IFR REG.
IFR
TMPD
BCC 1$ ;IF NO ERROR, PROCEED
ERROR ;ELSE, REPORT IT
SEV ;FLAG AN ERROR TO MAINLINE ROUTINE TRAP C$ERROR
RTS PC ; AND TAKE AN ABNORMAL RETURN

1$: MOV R0,-(SP) ;PRESERVE R0
MOVB TMPD,R0 ;PUT VALUE HERE TO PRESERVE TMPD
ROLB R0 ;'IRQ' GOES INTO CARRY BIT
ROLB R0 ;'T1' GOES INTO CARRY BIT
MOV (SP)+,R0 ;RESTORE R0
RTS PC
-----
: KICKT1 -- INIT. TIMER # 1 BY THE FOLLOWING PROCEDURE:
: READ T1L-H (ADDR 07) TO GET THE LAST VALUE LOADED INTO IT
: WRITE THAT VALUE INTO T1C-H (ADDR 05) TO RESET THE 'T1' INTERRUPT FLAG
: AND CAUSE THE RE-LOADING OF BOTH COUNTERS.
-----
KICKT1: MOV R3,-(SP) ;SAVE CALLER'S REGISTER CONTENTS
JSR R5,READ ;GET THE CURRENT SETTING OF THE HIGH LATCH
T1LH
TMP7
BCS 10$ ;IF ERROR, EXIT

MOV #T1CH,@SEL4 ;SETUP ADDRESS FOR M-LOOP WRITE
MOVB TMP7,@SEL6 ;SETUP DATA FOR SAME
MOVB TMP7,TMP5+1 ;PUT HERE TOO. BECAUSE WE'RE GOING TO WRITE IT.
BICB #IFRT1,@BSEL3 ;CLEAR THE INTERRUPT BIT -- JUST IN CASE
MOVB #WFILOC,@BSEL2 ;TELL THE M-LOOP TO WRITE THE BYTE FOR US
MOV #60,R3 ;SETUP TIMEOUT COUNTER
5$: BITB #MRDY,@BSEL2 ;WAIT FOR M-READY TO BE SET
BNE 10$ ;AS SOON AS 'MRDY' IS SET, EXIT!
SOB R3,5$ ;IF NO TIMEOUT, CHECK AGAIN FOR M-READY
GTDF E14,ERR4 ;ELSE, 'MRDY' TIMEOUT
; QUEUE 'DEVICE FATAL' ERROR # 84
MOV #T.EDF,ERRTYP
MOV #84,ERRNBR
MOV #EM4,ERRMSG
MOV #ERR4,ERRBLK

SEC ;INDICATE THE FAILURE & EXIT

10$: MOV (SP)+,R3 ;RESTORE REGISTER

```

CVDMAA.P11 12-DEC-80 15:59

TEST 27 -- VIA TIMER # 1 ONE-SHOT MODE

8288 036274 000207

RTS PC ;IMMEDIATE RETURN

8289

8290

8291

8292

8293

8294

8295

8296

8297

8298

8299

8300

8301 036276 004537 004064

8302 036302 120000

8303 036304 002440

8304 036306 103003

8305 036310

8306 036310 104460

8307 036312 000262

8308 036314 000207

8309

8310 036316 010046 002440

8311 036320 113700

8312 036324 106100

8313 036326 012600

8314 036330 000207

8315

```

-----
: GETPB7 -- PUT THE CURRENT SETTING OF 'PB7' (BIT 7 OF ORB W/IN THE VIA CHIP)
: INTO THE CARRY BIT SO IT CAN BE TESTED UPON RETURN.
:
: CALLING SEQUENCE:
:
: JSR PC,GETPB7
: <TEST FOR PB7 SET OR CLEARED WITH 'BCS' OR 'BCC' INSTR'S>
-----

```

```

GETPB7: JSR R5,READ ;GET THE REGISTER THAT CONTAINS 'PB7'
        ORB
        TMO
        BCC 1$ ;IF NO ERROR, PROCEED
        ERROR ;ELSE, REPORT IT
        SEV ;FLAG AN ERROR TO MAINLINE ROUTINE TRAP C$ERROR
        RTS PC ; AND TAKE AN ABNORMAL RETURN

1$: MOV R0,-(SP) ;PRESERVE THIS REGISTER FOR THE CALLER
    MOVB TMO,R0 ;PUT ITS CONTENTS HERE SO WE CAN MANIPULATE IT
    ROLB R0 ;PUT 'PB7' INTO THE CARRY BIT
    MOV (SP)+,R0 ;RESTORE R0 FOR THE CALLER
    RTS PC ;RETURN WITH 'PB7' IN THE CARRY BIT

```



CVDMAA.P11 12-DEC-80 15:59

TEST 28 -- VIA TIMER 1 FREE-RUNNING MODE TEST

.SBTTL TEST 28 -- VIA TIMER 1 FREE-RUNNING MODE TEST

8316  
8317  
8318  
8319  
8320  
8321  
8322  
8323  
8324  
8325  
8326  
8327  
8328  
8329  
8330  
8331  
8332  
8333  
8334  
8335  
8336  
8337  
8338  
8339  
8340  
8341  
8342  
8343  
8344  
8345  
8346  
8347  
8348  
8349  
8350  
8351  
8352  
8353  
8354  
8355  
8356  
8357  
8358  
8359  
8360  
8361  
8362  
8363  
8364  
8365  
8366  
8367  
8368  
8369  
8370  
8371

036332  
036332  
036332  
104402  
036334 004737 003762  
036340 103003

```

*****
*
* TEST 28 -- VIA TIMER 1 FREE-RUNNING MODE TEST
*
* THIS TEST VERIFIES THAT THE TIMER 1 COUNTER IS OPERATIONAL IN
* FREE-RUNNING MODE, IN EACH OF TWO SUBTESTS.
*
* THE PROGRAM PERIODICALLY CHECKS THE COUNTER TO VERIFY THAT:
*
* IT IS DECREMENTING AND EVENTUALLY REACHES 0,
*
* RELOADS FROM THE LATCHES, AND
*
* CONTINUES TO DECREMENT.
*
* IN THE FIRST SUBTEST, THE FOLLOWING IS PERFORMED :
*
* A MASTER CLEAR IS DONE AND THE TIMER IS PLACED IN FREE-RUNNING MODE
* BY SETTING ACR7 TO 0 & ACR6 TO 1 (MODE 1). AND THE PROGRAM CHECKS
* FOR THE 'T1' (BIT 6 IN IFR) TO BE INITIALLY CLEARED.
*
* THEN T1L-L (ADR 04) IS LOADED WITH 125 (OCTAL) AND T1C-H (ADR 05) IS
* LOADED WITH 125 (OCTAL) STARTING THE COUNTER.
*
* THE COUNT IS ALLOWED TO REACH 0 AGAIN, AND THE 'T1' IS READ AND
* CHECKED TO BE SET.
*
* T1C-H (ADR 05) IS READ AND 'T1' IS CHECKED TO BE STILL SET.
*
* THE COUNTER LO BYTE IS READ AND THE 'T1' IS READ AND CHECKED TO BE
* CLEARED BY THE READ OF T1C-L.
*
* THE COUNT IS ALLOWED TO REACH 0 ONCE MORE AND 'T1' IS CHECKED TO BE
* SET AGAIN.
*
* T1L-L IS LOADED WITH 252 (OCTAL) AND 'T1' IS CHECKED TO BE STILL
* SET.
*
* T1C-H IS LOADED WITH 252 (OCTAL) AND 'T1' IS READ AND CHECKED TO BE
* CLEARED BY THE LOADING OF T1C-H.
*
* IN THE SECOND SUBTEST, ALL OF THE ABOVE OPERATIONS ARE REPEATED, WITH
* ACR7 = 1, AND ACR6 = 1 (MODE 3). ALSO, PB7 IS VERIFIED FOR PROPER
* STATE AT THE PROPER TIME.

```

\*\*\*\*\*

```

: BGNTST
: BGNSUB
: JSR PC,MSTCLR ;INIT DMV & ENTER M-LOOP
: BCC 1$ ;IF NO ERROR, PROCEED WITH TESTING
: T28::
: T28.1: TRAP CSBSUB

```

CVDMAA.P11 12-DEC-80 15:59

TEST 28 -- VIA TIMER 1 FREE-RUNNING MODE TEST

```

8372 036342          EPROR          ;ELSE, REPORT ERROR
8373 036342 104460          TRAP      C$ERROR
8374 036344          ESCAPE TST          ; & EXIT TEST
8375 036344 104410          TRAP      C$ESCAPE
8376 036346 001514          .WORD    L10076-.
8377 036350 004537 004660 1$: JSR      R5,INITT1      ;INITIALIZE TIMER # 1
8378 036354 000000          ;          0 ==> LATCHES
8379 036356 000100          ;          MODE 1 & 'T1' INT. ENABLE FLAG CLEARED
8380 036360 103003          BCC      .+10      ;IF NO ERROR, PROCEED
8381 036362          ERROR          ;ELSE, REPORT IT
8382 036362 104460          TRAP      C$ERROR
8383 036364          ESCAPE TST          ;          AND EXIT THIS TEST
8384 036364 104410          TRAP      C$ESCAPE
8385 036366 001474          .WORD    L10076-.
8386 036370 004737 036112 JSR      PC,GETT1      ;IS 'T1' SET?
8387 036374 102002          BVC      .+6      ;IF NO ERROR, PROCEED
8388 036376          ESCAPE SUB          ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
8389 036376 104410          TRAP      C$ESCAPE
8390 036400 000414          .WORD    L10077-.
8391 036402 103006          BCC      2$
8392 036404          GEDF      EM50A,ERR50 ;NO, GOOD.
8393          ;          YES, REPORT IT'S NOT BEING CLEARED @ INIT.
8394          ;          'DEVICE FATAL' ERROR # 85
8394 036404 104455          TRAP      C$ERDF
8395 036406 000125          .WORD    85
8396 036410 016067          .WORD    EM50A
8397 036412 010762          .WORD    ERR50
8398 036414          ESCAPE SUB          ; & EXIT TEST
8399 036414 104410          TRAP      C$ESCAPE
8400 036416 000376          .WORD    L10077-.
8401
8402 ;-----
8403
8404 036420 004537 036060 2$: JSR      R5,LODT1C      ;RELOAD TIMER 1'S COUNTERS WITH NEW VALUES:
8405 036424          .BYTE    125,125
8406
8407 ;-----
8408
8409 036426 005003 036112 3$: CLR      R3          ;INITIALIZE TIMEOUT COUNTER
8410 036430 004737          JSR      PC,GETT1      ;'T1' SHOULD BE SET. IS IT?
8411 036434 102002          BVC      .+6      ;IF NO ERROR, PROCEED
8412 036436          ESCAPE SUB          ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
8413 036436 104410          TRAP      C$ESCAPE
8414 036440 000354          .WORD    L10077-.
8415 036442 103407          BCS      4$
8416 036444 077307          SOB      R3,3$      ;YES, GOOD.
8417 036446          GEDF      EM50F,ERR50 ;NO, IF NO TIMEOUT, LOOK AGAIN
8418          ;          ELSE, SAY IT WASN'T SET BY T1 TIMEOUT
8419          ;          'DEVICE FATAL' ERROR # 86
8419 036446 104455          TRAP      C$ERDF
8420 036450 000126          .WORD    86
8421 036452 016341          .WORD    EM50F
8422 036454 010762          .WORD    ERR50
8423 036456          ESCAPE SUB          ;IF ERROR, THE REST OF THIS TEST IS UN-DOABLE!
8424 036456 104410          TRAP      C$ESCAPE
8425 036460 000334          .WORD    L10077-.
8426
8427 036462 004537 004064 4$: JSR      R5,READ      ;READING T1CH SHOULDN'T CLEAR 'T1'

```



CVDMAA.P11 12-DEC-80 15:59

TEST 28 -- VIA TIMER 1 FREE-RUNNING MODE TEST

```

8428 036466 120005          T1CH
8429 036470 002452          TMP5          ; (WE DON'T CARE WHAT THIS IS)
8430 036472 103003          BCC          .+10      ; IF NO ERROR, PROCEED
8431 036474              ERROR          ; ELSE, REPORT IT
8432 036474 104460          ESCAPE TST          ; AND EXIT THIS TEST          TRAP    C$ERROR
8433 036476              ;                                     ;                               TRAP    C$ESCAPE
8434 036476 104410          ;                                     ;                               .WORD  L10076-.
8435 036500 001362          JSR    PC,GETT1      ; CHECK 'T1' -- IT SHOULD STILL BE SET
8436 036502 004737 036112  BVC    .+6          ; IF NO ERROR, PROCEED
8437 036506 102002          ESCAPE SUB          ; ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
8438 036510              ;                                     ;                               TRAP    C$ESCAPE
8439 036510 104410          ;                                     ;                               .WORD  L10077-.
8440 036512 000302          BCS    6$          ; IT IS, GOOD.
8441 036514 103404          GEDF   EM50G,ERR50 ; CLEARED BY READING T1CH!!
8442 036516              ; 'DEVICE FATAL' ERROR # 87
8443              ;                                     ;                               TRAP    C$ERDF
8444 036516 104455          ;                                     ;                               .WORD  87
8445 036520 000127          ;                                     ;                               .WORD  EM50G
8446 036522 016406          ;                                     ;                               .WORD  ERR50
8447 036524 010762          6$: JSR    PC,KICKT1  ; KICK IT OFF AGAIN SO WE CAN PRESERVE TIMING
8448 036526 004737 036150  BCC    .+10      ; IF NO ERROR, PROCEED
8449 036532 103003          ERROR          ; ELSE, REPORT IT
8450 036534              ;                                     ;                               TRAP    C$ERROR
8451 036534 104460          ESCAPE TST          ; AND EXIT THIS TEST          TRAP    C$ESCAPE
8452 036536              ;                                     ;                               .WORD  L10076-.
8453 036536 104410          ;WAIT FOR IT TO FINISH:
8454 036540 001322          CLR    R3          ; INITIALIZE TIMEOUT COUNTER
8455              JSR    PC,GETT1      ; 'T1' SHOULD BE SET. IS IT?
8456              BVC    .+6          ; IF NO ERROR, PROCEED
8457              ESCAPE SUB          ; ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
8458 036542 005003 036112  7$:          ;                                     ;                               TRAP    C$ESCAPE
8459 036544 004737          ;                                     ;                               .WORD  L10077-.
8460 036550 102002          BCS    8$          ; YES, GOOD.
8461 036552 104410          SOB    R3,7$       ; NO, IF NO TIMEOUT, LOOK AGAIN
8462 036554 000240          BR    10$          ; IF TIMEOUT, BYPASS NEXT CHECK (THIS DONE ABOVE)
8463 036556 103402          JSR    R5,READ     ; READING T1CL SHOULD CLEAR 'T1'
8464 036560 077307          ;                                     ;                               TRAP    C$ERROR
8465 036562 000422          ;                                     ;                               .WORD  L10076-.
8466 036564 004537 004064  8$: JSR    R5,READ     ; (WE DON'T CARE WHAT THIS IS EITHER)
8467 036570 120004          T1CL          ; IF NO ERROR, PROCEED
8468 036572 002450          TMP4          ; ELSE, REPORT IT
8469 036574 103003          BCC    .+10      ; AND EXIT THIS TEST          TRAP    C$ERROR
8470 036576 104460          ESCAPE TST          ;                                     ;                               TRAP    C$ESCAPE
8471 036576 104410          ;                                     ;                               .WORD  L10076-.
8472 036600 104410          JSR    PC,GETT1      ; CHECK 'T1' -- IT SHOULD BE CLEARED NOW
8473 036600 104410          BVC    .+6          ; IF NO ERROR, PROCEED
8474 036602 001260          ESCAPE SUB          ; ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
8475 036604 004737 036112  ;                                     ;                               TRAP    C$ESCAPE
8476 036610 102002          ;                                     ;                               .WORD  L10077-.
8477 036612 104410          BCC    10$          ; IT IS, GOOD.
8478 036612 104410          GEDF   EM50C,ERR50 ; NOT CLEARED! REPORT IT.
8479 036614 000200          ; 'DEVICE FATAL' ERROR # 88
8480 036616 103004          ;                                     ;                               TRAP    C$ERDF
8481 036620 104455          ;                                     ;                               .WORD  87
8482              ;                                     ;                               .WORD  EM50G
8483              ;                                     ;                               .WORD  ERR50

```

CVDMAA.P11 12-DEC-80 15:59

TEST 28 -- VIA TIMER 1 FREE-RUNNING MODE TEST

8484	036622	000130							.WORD	88
8485	036624	016203							.WORD	EM50C
8486	036626	010762							.WORD	ERR50
8487	036630	005003								
8488	036632	004737	036112	10\$:	CLR	R3		;RE-INITIALIZE THE TIMEOUT COUNTER		
8489	036636	102002		12\$:	JSR	PC,GETT1		;WAIT FOR 'T1' TO GET SET AGAIN		
8490	036640				BVC	.+6		;IF NO ERROR, PROCEED		
8491	036640	104410			ESCAPE	SUB		;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT		
8492	036642	000152							TRAP	C\$ESCAPE
8493	036644	103407							.WORD	L10077-.
8494	036646	077307			BCS	14\$		;GOT IT -- GOOD.		
8495	036650				SOB	R3,12\$		;NOT YET. IF NO TIMEOUT, TRY AGAIN.		
8496					GEDF	EM50X,ERR50		;ELSE, REPORT 'T1' NOT RESET		
8497	036650	104455						; 'DEVICE FATAL' ERROR # 89		
8498	036652	000131							TRAP	C\$ERDF
8499	036654	017354							.WORD	89
8500	036656	010762							.WORD	EM50X
8501	036660				ESCAPE	SUB		;IF ERROR, CAN'T CONTINUE THIS TEST	.WORD	ERR50
8502	036660	104410							TRAP	C\$ESCAPE
8503	036662	000132							.WORD	L10077-.
8504	036664	112737	000252	002455	14\$:	MOVB	#252,TMP6+1	;SETUP FOR AND		
8505	036672	004537	004310		JSR	R5,WRITE		; LOAD T1LL (ADDR 6)		
8506	036676	120006			T1LL					
8507	036700	002455			TMP6+1			; WITH 252 OCTAL		
8508	036702	103003			BCC	.+10		;IF NO ERROR, PROCEED		
8509	036704				ERROR			;ELSE, REPORT IT		
8510	036704	104460							TRAP	C\$ERROR
8511	036706				ESCAPE	TST		; AND EXIT THIS TEST		
8512	036706	104410							TRAP	C\$ESCAPE
8513	036710	001152							.WORD	L10076-.
8514	036712	004737	036112		JSR	PC,GETT1		;THIS SHOULDN'T CLEAR 'T1'		
8515	036716	102002			BVC	.+6		;IF NO ERROR, PROCEED		
8516	036720				ESCAPE	SUB		;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT		
8517	036720	104410							TRAP	C\$ESCAPE
8518	036722	000072							.WORD	L10077-.
8519	036724	103406			BCS	16\$		;IT DIDN'T -- GOOD.		
8520	036726				GEDF	EM50M,ERR50		;WOOPS, IT DID!! REPORT FAILURE		
8521								; 'DEVICE FATAL' ERROR # 90		
8522	036726	104455							TRAP	C\$ERDF
8523	036730	000132							.WORD	90
8524	036732	017012							.WORD	EM50M
8525	036734	010762							.WORD	ERR50
8526	036736				ESCAPE	SUB		; THE REST OF THIS TEST IS INVALID TOO!		
8527	036736	104410							TRAP	C\$ESCAPE
8528	036740	000054							.WORD	L10077-.
8529	036742	112737	000252	002457	16\$:	MOVB	#252,TMP7+1	;SETUP FOR AND		
8530	036750	004537	004310		JSR	R5,WRITE		; LOAD T1LH (ADDR 7)		
8531	036754	120007			T1LH					
8532	036756	002457			TMP7+1			; WITH 252 OCTAL		
8533	036760	103003			BCC	.+10		;IF NO ERROR, PROCEED		
8534	036762				ERROR			;ELSE, REPORT IT		
8535	036762	104460							TRAP	C\$ERROR
8536	036764				ESCAPE	TST		; AND EXIT THIS TEST		
8537	036764	104410							TRAP	C\$ESCAPE
8538	036766	001074							.WORD	L10076-.
8539	036770	004737	036112		JSR	PC,GETT1		;THIS SHOULD CLEAR 'T1'		



CVDMAA.P11 12-DEC-80 15:59

TEST 28 -- VIA TIMER 1 FREE-RUNNING MODE TEST

```

8540 036774 102002          BVC      .+6          ;IF NO ERROR, PROCEED
8541 036776                ESCAPE  SUB          ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
8542 036776 104410                TRAP      C$ESCAPE
8543 037000 000014                .WORD    L10077-.
8544 037002 103004          BCC      18$          ;IT DID -- GOOD.
8545 037004                GEDF    EM50A,ERR50 ;NOP! REPORT: "T1" NOT CLEARED BY LOADING T1LH
8546                                ;          'DEVICE FATAL' ERROR # 91
8547 037004 104455                TRAP      C$ERDF
8548 037006 000133                .WORD    91
8549 037010 016067                .WORD    EM50A
8550 037012 010762                .WORD    ERR50
8551 037014
8552 037014 18$:          ;THAT'S ALL FOLKS!
8553 037014                ENDSUB
8554 037014 104403                L10077: TRAP      C$ESUB
8555                                ;-----
8556 037016                BGNSUB
8557 037016                T28.2: TRAP      C$BSUB
8558 037016 104402          JSR      PC,MSTCLR    ;INIT DMV & ENTER M-LOOP
8559 037020 004737 003762          BCC      1$          ;IF NO ERROR, PROCEED WITH TESTING
8560 037024 103003          ERROR
8561 037026                ;ELSE, REPORT ERROR
8562 037026 104460                TRAP      C$ERROR
8563 037030          ESCAPE  TST          ; & EXIT TEST
8564 037030 104410                TRAP      C$ESCAPE
8565 037032 001030                .WORD    L10076-.
8566 037034 112737 000377 002445 1$: MOVB    #377,TMP2+1 ;SETUP DDRB SUCH THAT ORB IS AN INPUT/OUTPUT REG
8567 037042 004537 004310          JSR      R5,WRITE
8568 037046 120002          DDRB
8569 037050 002445          TMP2+1
8570 037052 103003          BCC      .+10       ;IF NO ERROR, PROCEED
8571 037054                ERROR          ;ELSE, REPORT IT
8572 037054 104460                TRAP      C$ERROR
8573 037056          ESCAPE  TST          ; AND EXIT THIS TEST
8574 037056 104410                TRAP      C$ESCAPE
8575 037060 001002                .WORD    L10076-.
8576 037062 112737 000030 002441 MOVB    #30,TMP0+1 ;CLEAR ALL BITS IN ORB EXCEPT DTR L & RTS L
8577 037070 004537 004310          JSR      R5,WRITE ; BY DOING THIS, WE SHOULD EXPECT PB7 TO BE
8578 037074 120000          ORB          ; CLEARED IF MODE 3 DOESN'T WORK PROPERLY.
8579 037076 002441          TMP0+1
8580 037100 103003          BCC      .+10       ;IF NO ERROR, PROCEED
8581 037102                ERROR          ;ELSE, REPORT IT
8582 037102 104460                TRAP      C$ERROR
8583 037104          ESCAPE  TST          ; AND EXIT THIS TEST
8584 037104 104410                TRAP      C$ESCAPE
8585 037106 000754                .WORD    L10076-.
8586 037110 004537 004660          JSR      R5,INITT1 ;INITIALIZE TIMER # 1
8587 037114 000000          0          ; 0 ==> LATCHES
8588 037116 000300          BIT7+BIT6 ; MODE 3 & 'T1' INT. ENABLE FLAG CLEARED
8589 037120 103003          BCC      .+10       ;IF NO ERROR, PROCEED
8590 037122                ERROR          ;ELSE, REPORT IT
8591 037122 104460                TRAP      C$ERROR
8592 037124          ESCAPE  TST          ; AND EXIT THIS TEST
8593 037124 104410                TRAP      C$ESCAPE
8594 037126 000734                .WORD    L10076-.
8595 037130 004737 036112          JSR      PC,GETT1 ;IS 'T1' SET?

```

CVDMAA.P11 12-DEC-80 15:59

TEST 28 -- VIA TIMER 1 FREE-RUNNING MODE TEST

```

8596 037134 102002          BVC      .+6      ;IF NO ERROR, PROCEED
8597 037136                ESCAPE  SUB      ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
8598 037136 104410                TRAP    C$ESCAPE
8599 037140 000720                .WORD  L10100-.
8600 037142 103006          BCC      2$
8601 037144                GEDF   EM50A,ERR50 ;NO, GOOD.
8602                ;YES, REPORT IT'S NOT BEING CLEARED @ INIT.
8603 037144 104455                ;      'DEVICE FATAL' ERROR # 92
8604 037146 000134                TRAP    C$ERDF
8605 037150 016067                .WORD  92
8606 037152 010762                .WORD  EM50A
8607 037154                .WORD  ERR50
8608 037154 104410          ESCAPE  SUB      ;      & EXIT TEST
8609 037156 000702                TRAP    C$ESCAPE
8610                .WORD  L10100-.
8611                ;-----
8612                ;
8613 037160 004537 036060      2$:    JSR      R5,LODT1C ;RELOAD TIMER 1'S COUNTERS WITH NEW VALUES:
8614 037164      125      125      .BYTE  125,125
8615                ;-----
8616                ;
8617                ;
8618 037166 005003 036112      4$:    CLR      R3      ;INITIALIZE TIMEOUT COUNTER
8619 037170 004737                JSR      PC,GETT1 ;'T1' SHOULD BE SET. IS IT?
8620 037174 102002          BVC      .+6      ;IF NO ERROR, PROCEED
8621 037176                ESCAPE  SUB      ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
8622 037176 104410                TRAP    C$ESCAPE
8623 037200 000660                .WORD  L10100-.
8624 037202 103407          BCS      5$
8625 037204 077307          SOB      R3,4$   ;YES, GOOD.
8626 037206                GEDF   EM50F,ERR50 ;NO, IF NO TIMEOUT, LOOK AGAIN
8627                ;ELSE, SAY IT WASN'T SET BY T1 TIMEOUT
8628 037206 104455                ;      'DEVICE FATAL' ERROR # 93
8629 037210 000135                TRAP    C$ERDF
8630 037212 016341                .WORD  93
8631 037214 010762                .WORD  EM50F
8632 037216                .WORD  ERR50
8633 037216 104410          ESCAPE  SUB      ;IF ERROR, THE REST OF THIS TEST IS UN-DOABLE!
8634 037220 000640                TRAP    C$ESCAPE
8635                .WORD  L10100-.
8636                ;-----
8637                ;
8638 037222 004737 036276      5$:    JSR      PC,GETPB7 ;GET 'PB7'. IS IT SET?
8639 037222 102002          BVC      .+6      ;IF NO ERROR, PROCEED
8640 037226                ESCAPE  SUB      ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
8641 037230                TRAP    C$ESCAPE
8642 037230 104410                .WORD  L10100-.
8643 037232 000626          BCS      36$
8644 037234 103406          GEDF   EM50U,ERR50 ;YES, GOOD.
8645 037236                ;NO, REPORT IT NOT SET.
8646                ;      'DEVICE FATAL' ERROR # 94
8647 037236 104455                TRAP    C$ERDF
8648 037240 000136                .WORD  94
8649 037242 017174                .WORD  EM50U
8650 037244 010762                .WORD  ERR50
8651 037246                ESCAPE  SUB      ;      & ALLOW RESTART OF THIS SUBTEST

```



CVDMAA.P11 12-DEC-80 15:59

TEST 28 -- VIA TIMER 1 FREE-RUNNING MODE TEST

```

8652 037246 104410
8653 037250 000610
8654
8655
8656
8657 037252 004537 004064
8658 037256 120005
8659 037260 002452
8660 037262 103003
8661 037264
8662 037264 104460
8663 037266
8664 037266 104410
8665 037270 000572
8666 037272 004737 036112
8667 037276 102002
8668 037300
8669 037300 104410
8670 037302 000556
8671 037304 103406
8672 037306
8673
8674 037306 104455
8675 037310 000137
8676 037312 016406
8677 037314 010762
8678 037316
8679 037316 104410
8680 037320 000540
8681
8682
8683
8684 037322 005003
8685 037324 004737 036112
8686 037330 102002
8687 037332
8688 037332 104410
8689 037334 000524
8690 037336 103407
8691 037340 077307
8692 037342
8693
8694 037342 104455
8695 037344 000140
8696 037346 016730
8697 037350 010762
8698 037352
8699 037352 104410
8700 037354 000504
8701
8702
8703
8704 037356
8705 037356 004737 036276
8706 037362 102002
8707 037364

```

```

TRAP C$ESCAPE
.WORD L10100-.

-----
36$: JSR R5,READ ;READING T1CH SHOULDN'T CLEAR 'T1'
T1CH ;
TMP5 ; (WE DON'T CARE WHAT THIS IS)
BCC .+10 ;IF NO ERROR, PROCEED
ERROR ;ELSE, REPORT IT
ESCAPE TST ; AND EXIT THIS TEST
TRAP C$ERROR
.WORD C$ESCAPE
L10076-.
JSR PC,GETT1 ;CHECK 'T1' -- IT SHOULD STILL BE SET
BVC .+6 ;IF NO ERROR, PROCEED
ESCAPE SUB ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
TRAP C$ESCAPE
.WORD L10100-.
BCS 37$ ;IT IS, GOOD.
GEDF EM50G,ERR50 ;Cleared BY READING T1CH!!
; 'DEVICE FATAL' ERROR # 95
TRAP C$ERDF
.WORD 95
.WORD EM50G
.WORD ERR50
ESCAPE SUB ; ALLOW RESTART OF THIS SUBTEST
TRAP C$ESCAPE
.WORD L10100-.

-----
37$: CLR R3 ;INITIALIZE TIMEOUT COUNTER AGAIN
38$: JSR PC,GETT1 ;WAIT FOR 'T1' TO BE SET AGAIN
BVC .+6 ;IF NO ERROR, PROCEED
ESCAPE SUB ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
TRAP C$ESCAPE
.WORD L10100-.
BCS 39$ ;GOT IT -- NO CHECK PB7
SOB R3,38$ ;NOT YET. IF NO TIMEOUT, LOOK AGAIN.
GEDF EM50L,ERR50 ;ELSE, TIMER NOT REALLY WORKING RIGHT!
; 'DEVICE FATAL' ERROR # 96
TRAP C$ERDF
.WORD 96
.WORD EM50L
.WORD ERR50
ESCAPE SUB
TRAP C$ESCAPE
.WORD L10100-.

-----
39$: JSR PC,GETPB7 ;GET 'PB7'. IS IT SET?
BVC .+6 ;IF NO ERROR, PROCEED
ESCAPE SUB ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT

```

CVDMAA.P11 12-DEC-80 15:59

TEST 28 -- VIA TIMER 1 FREE-RUNNING MODE TEST

```

8708 037364 104410
8709 037366 000472
8710 037370 103404
8711 037372
8712
8713 037372 104455
8714 037374 000141
8715 037376 017471
8716 037400 010762
8717
8718
8719
8720 037402 004737 036150
8721 037406 103003
8722 037410
8723 037410 104460
8724 037412
8725 037412 104410
8726 037414 000446
8727
8728
8729 037416 005003
8730 037420 004737 036112
8731 037424 102002
8732 037426
8733 037426 104410
8734 037430 000430
8735 037432 103402
8736 037434 077307
8737 037436 000442
8738 037440 004537 004064
8739 037444 120004
8740 037446 002450
8741 037450 103003
8742 037452
8743 037452 104460
8744 037454
8745 037454 104410
8746 037456 000404
8747 037460 004737 036112
8748 037464 102002
8749 037466
8750 037466 104410
8751 037470 000370
8752 037472 103006
8753 037474
8754
8755 037474 104455
8756 037476 000142
8757 037500 016203
8758 037502 010762
8759 037504
8760 037504 104410
8761 037506 000352
8762 037510 005003
8763 037512 004737 036112

```

```

BCS 6$
GEDF EM50Z,ERR50
;YES, GOOD.
;NO, REPORT 'PB7' NOT SET AFTER SECOND CYCLE
; 'DEVICE FATAL' ERROR # 97
TRAP C$ESCAPE
.WORD L10100-.

-----
6$: JSR PC,KICKT1 ;KICK IT OFF AGAIN SO WE CAN PRESERVE TIMING
BCC .+10 ;IF NO ERROR, PROCEED
ERROR ;ELSE, REPORT IT
ESCAPE TST ; AND EXIT THIS TEST
TRAP C$ERROR
.WORD C$ESCAPE
L10076-.

;WAIT FOR IT TO FINISH:
7$: CLR R3 ;INITIALIZE TIMEOUT COUNTER
JSR PC,GETT1 ;'T1' SHOULD BE SET. IS IT?
BVC .+6 ;IF NO ERROR, PROCEED
ESCAPE SUB ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
TRAP C$ESCAPE
.WORD L10100-.

BCS 8$
SOB R3,7$ ;YES, GOOD.
BR 14$ ;NO, IF NO TIMEOUT, LOOK AGAIN
8$: JSR R5,READ ;IF TIMEOUT, BYPASS NEXT CHECK (THIS DONE ABOVE)
T1CL ;READING T1CL SHOULD CLEAR 'T1'
TMP4 ; (WE DON'T CARE WHAT THIS IS EITHER)
BCC .+10 ;IF NO ERROR, PROCEED
ERROR ;ELSE, REPORT IT
ESCAPE TST ; AND EXIT THIS TEST
TRAP C$ERROR
.WORD C$ESCAPE
L10076-.

JSR PC,GETT1 ;CHECK 'T1' -- IT SHOULD BE CLEARED NOW
BVC .+6 ;IF NO ERROR, PROCEED
ESCAPE SUB ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
TRAP C$ESCAPE
.WORD L10100-.

BCC 9$
GEDF EM50C,ERR50 ;IT IS, GOOD.
;NOT CLEARED! REPORT IT.
; 'DEVICE FATAL' ERROR # 98
TRAP C$ERDF
.WORD 98
.WORD EM50C
.WORD ERR50
ESCAPE SUB ;IF THIS ERROR OCCURED, EXIT SUBTEST
TRAP C$ESCAPE
.WORD L10100-.

9$: CLR R3 ;RE-INITIALIZE THE TIMEOUT COUNTER
12$: JSR PC,GETT1 ;WAIT FOR 'T1' TO GET SET AGAIN

```



CVDMAA.P11 12-DEC-80 15:59

TEST 28 -- VIA TIMER 1 FREE-RUNNING MODE TEST

```

8764 037516 102002           BVC      .+6          ;IF NO ERROR, PROCEED
8765 037520                  ESCAPE  SUB          ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
8766 037520 104410                               TRAP    C$ESCAPE
8767 037522 000336                               .WORD  L10100-.
8768 037524 103407           BCS      14$         ;GOT IT -- GOOD.
8769 037526 077307           SOB      R3,12$      ;NOT YET. IF NO TIMEOUT, TRY AGAIN.
8770 037530                  GEDF    EM50X,ERR50 ;ELSE, REPORT 'T1' NOT RESET
8771                                   ;          'DEVICE FATAL' ERROR # 99
8772 037530 104455                               TRAP    C$ERDF
8773 037532 000143                               .WORD  99
8774 037534 017354                               .WORD  EM50X
8775 037536 010762                               .WORD  ERR50
8776 037540                  ESCAPE  SUB          ;IF ERROR, CAN'T CONTINUE THIS TEST
8777 037540 104410                               TRAP    C$ESCAPE
8778 037542 000316                               .WORD  L10100-.
8779 037544 112737 000252 002455 14$:    MOVB    #252,TMP6+1 ;SETUP FOR AND
8780 037552 004537 004310      JSR      R5,WRITE   ; LOAD T1LL (ADDR 6)
8781 037556 120006      T1LL
8782 037560 002455      TMP6+1
8783 037562 103003      BCC      .+10      ; WITH 252 OCTAL
8784 037564                  ERROR          ;IF NO ERROR, PROCEED
8785 037564 104460                  ERROR          ;ELSE, REPORT IT
8786 037566                  ESCAPE  TST          ; AND EXIT THIS TEST
8787 037566 104410                               TRAP    C$ERROR
8788 037570 000272                               .WORD  C$ESCAPE
8789 037572 004737 036112      JSR      PC,GETT1   ;THIS SHOULDN'T CLEAR 'T1'
8790 037576 102002      BVC      .+6          ;IF NO ERROR, PROCEED
8791 037600                  ESCAPE  SUB          ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
8792 037600 104410                               TRAP    C$ESCAPE
8793 037602 000256                               .WORD  L10100-.
8794 037604 103406           BCS      16$         ;IT DIDN'T -- GOOD.
8795 037606                  GEDF    EM50M,ERR50 ;WOOPS, IT DID!! REPORT FAILURE
8796                                   ;          'DEVICE FATAL' ERROR # 100
8797 037606 104455                               TRAP    C$ERDF
8798 037610 000144                               .WORD  100
8799 037612 017012                               .WORD  EM50M
8800 037614 010762                               .WORD  ERR50
8801 037616                  ESCAPE  SUB          ; THE REST OF THIS TEST IS INVALID TOO!
8802 037616 104410                               TRAP    C$ESCAPE
8803 037620 000240                               .WORD  L10100-.
8804 037622 004737 036276      JSR      PC,GETPB7 ;'PB7' SHOULD BE LOW HERE
8805 037626 102002      BVC      .+6          ;IF NO ERROR, PROCEED
8806 037630                  ESCAPE  SUB          ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
8807 037630 104410                               TRAP    C$ESCAPE
8808 037632 000226                               .WORD  L10100-.
8809 037634 103054           BCC      17$         ;IT WASN'T, GOOD.
8810 037636 004537 004064      JSR      R5,READ   ;GET IFR FOR ERROR MESSAGE
8811 037642 120015      IFR
8812 037644 002472      TMPD
8813 037646 103003      BCC      .+10      ;IF NO ERROR, PROCEED
8814 037650                  ERROR          ;ELSE, REPORT IT
8815 037650 104460                  ERROR          ; AND EXIT THIS TEST
8816 037652                  ESCAPE  TST          ; AND EXIT THIS TEST
8817 037652 104410                               TRAP    C$ERROR
8818 037654 000206                               .WORD  C$ESCAPE
8819 037656 004537 004064      JSR      R5,READ   ;GET T1CL FOR ERROR MESSAGE

```





CVDMAA.P11

12-DEC-80 15:59

TEST 28 -- VIA TIMER 1 FREE-RUNNING MODE TEST

```

8876 040014 004737 036112      JSR   PC,GETT1      ;THIS SHOULD CLEAR 'T1'
8877 040020 102002              BVC   .+6           ;IF NO ERROR, PROCEED
8878 040022                      ESCAPE SUB           ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
8879 040022 104410              TRAP  C$ESCAPE
8880 040024 000034              .WORD L10100-.
8881 040026 103006              BCC   18$           ;IT DID -- GOOD.
8882 040030                      GEDF  EM50A,ERR50  ;NOP! REPORT: 'T1' NOT CLEARED BY LOADING T1LH
8883                                ; 'DEVICE FATAL' ERROR # 102
8884 040030 104455              TRAP  C$ERDF
8885 040032 000146              .WORD 102
8886 040034 016067              .WORD EM50A
8887 040036 010762              .WORD ERR50
8888 040040                      ESCAPE SUB
8889 040040 104410              TRAP  C$ESCAPE
8890 040042 000016              .WORD L10100-.
8891 040044 004537 004660      18$: JSR   R5,INITT1  ;RE-INITIALIZE IT TO STOP ITS FUNCTIONING
8892 040050 000001              1
8893 040052 000000              0
8894 040054 103001              BCC   .+4           ;IF NO ERROR, EXIT
8895 040056                      ERROR              ;ELSE, REPORT IT
8896 040056 104460              TRAP  C$ERROR
8897                                ;THAT'S ALL FOLKS!
8898 040060                      ENDSUB
8899 040060                      L10100: TRAP  C$ESUB
8900 040060 104403                      L10076: TRAP  C$SETST
8901 040062                      ENDTST
8902 040062
8903 040062 104401
8904
8905
8906

```

CVDMAA.P11 12-DEC-80 15:59

HARDWARE PARAMETER CODING SECTION

.SBTTL HARDWARE PARAMETER CODING SECTION

8907  
8908  
8909  
8910  
8911  
8912  
8913  
8914  
8915  
8916  
8917  
8918  
8919  
8920  
8921  
8922  
8923  
8924  
8925  
8926  
8927  
8928  
8929  
8930  
8931  
8932  
8933  
8934  
8935  
8936  
8937  
8938  
8939  
8940  
8941  
8942  
8943  
8944

040064  
040064 000015  
040066  
040066  
040066 000031  
040070 040120  
040072 160020  
040074 177776  
040076  
040076 001031  
040100 040146  
040102 000000  
040104 000674  
040106  
040106 002032  
040110 040177  
040112 007000  
040114 000000  
040116 000007  
040120  
040120  
040120  
042504 044526 042503  
040146 042504 044526 042503  
040177 104 053105 041511

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

BGNHRD

GPRMA ADDRES,0,0,160020,177776,YES

GPRMA VECTOR,2,0,0,674,YES

GPRMD PRIRTY,4,0,7000,0,7,YES

ENDHRD

.NLIST BEX  
ADDRESS: .ASCIZ /DEVICE CSR ADDRESS : /  
VECTOR: .ASCIZ /DEVICE VECTOR ADDRESS : /  
PRIRTY: .ASCIZ /DEVICE PRIORITY LEVEL : /  
.LIST BEX  
.EVEN

.WORD L10101-LSHARD/2  
LSHARD::

.WORD TSCODE  
.WORD ADDRESS  
.WORD TSLOLIM  
.WORD TSHILIM

.WORD TSCODE  
.WORD VECTOR  
.WORD TSLOLIM  
.WORD TSHILIM

.WORD TSCODE  
.WORD PRIRTY  
.WORD 7000  
.WORD TSLOLIM  
.WORD TSHILIM

.EVEN  
L10101:

8945



CVDMAA.P11 12-DEC-80 15:59

SOFTWARE PARAMETER CODING SECTION

.SBTTL SOFTWARE PARAMETER CODING SECTION

8946  
8947  
8948  
8949  
8950  
8951  
8952  
8953  
8954  
8955  
8956  
8957  
8958  
8959 040230  
8960 040230 000000  
8961 040232  
8962 040232  
8963  
8964 040232

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

BGNSFT

ENDSFT

.WORD L10102-L\$\$SOFT/2  
L\$\$SOFT::  
.EVEN  
L10102:

CVDMAA.P11 12-DEC-80 15:59

PATCH AREA FOR DEBUG

```

8965
8966 040232
8967          040332
8968 040332 000240
8969 040334 000240
8970 040336 000240
8971
8972
8973
8974
8975 040340
8976 040340
8977
8978 040340 000000
8979 040342 000000
8980 040344
8981          000001

```

```

.SBTTL PATCH AREA FOR DEBUG
PATCH:

```

```

.=.+100
NOP
NOP
NOP

```

\*\*\*\*\*

```

.SBTTL 'ENDMOD' & 'LASTAD'
      ENDMOD
      LASTAD

```

```

L$LAST::
.END

```

```

.EVEN
.WORD 0
.WORD 0

```





CVDMAA.P11 12-DEC-80 15:59

CROSS REFERENCE TABLE -- USER SYMBOLS

BSL4	002362	1789#	2185	2408																
BSL5	002364	1791#	2409																	
BSL6	002366	1792#	2181	2280	2410															
BSL7	002370	1794#	2411																	
BSLT0 =	000020	1625#																		
BSLT1 =	000021	1626#																		
BSLT2 =	000022	1628#																		
BSLT3 =	000023	1629#																		
BSLT4 =	000024	1631#																		
BSLT5 =	000025	1632#																		
BSLT6 =	000026	1634#																		
BSLT7 =	000027	1635#																		
BSR0	002246	1692#	2404*	3365																
BSR1	002250	1694#	2405*	3363																
BSR10	002266	1707#	2412*	3407																
BSR11	002270	1708#	2413*	3405																
BSR12	002272	1709#	2414*	3403																
BSR13	002274	1710#	2415*	3401																
BSR14	002276	1711#	2416*	3428																
BSR15	002300	1712#	2417*	3426																
BSR16	002302	1713#	2418*	3424																
BSR17	002304	1714#	2419*	3422																
BSR2	002252	1696#	2406*	3361																
BSR3	002254	1698#	2407*	3359																
BSR4	002256	1700#	2408*	3386																
BSR5	002260	1702#	2409*	3384																
BSR6	002262	1704#	2410*	3382																
BSR7	002264	1706#	2411*	3380																
BT1 =	003122	2095#	2805	2903	5184	5201														
BT2 =	003206	2096#	2835	2933	5202															
BUFARE	003122	2072#	2078	2095	2096															
CONSOL	002346	1737#	3551*	3642*																
CONTIN	020074	3545	3613#																	
CONTST	020174	3554	3642#																	
CSREGS=	000020	1521#	3960	3970	3992	4095														
C\$AU =	000052	1242#	3744																	
C\$AUTO=	000061	1242#	3696																	
C\$BRK =	000022	1242#	4586	4601	4933															
C\$BSEG=	000004	1242#	5623	5790	5862															
C\$BSUB=	000002	1242#	4267	4320	5372	5427	5493	5614	5774	5846	5950	6005	6060	6121						
		6719	7419	8369	8558															
C\$CEFG=	000045	1242#																		
C\$CLCK=	000062	1242#																		
C\$CLEA=	000012	1242#	3716																	
C\$CLOS=	000035	1242#																		
C\$CLP1=	000006	1242#																		
C\$CVEC=	000036	1242#	3562	3687	3710	3713	3795													
C\$DCLN=	000044	1242#																		
C\$DODU=	000051	1242#	3692																	
C\$DRPT=	000024	1242#																		
C\$DU =	000053	1242#	3731																	
C\$EDIT=	000003	1242#	1316																	
C\$ERDF=	000055	1242#	2587	2621	3802	3907	4012	4085	4153	4217	4230	4310	4332	4382						
		4435	4494	4511	4527	4956	5001	5219	5256	5277	5309	5330	5642	5663						
		5686	5712	5729	5752	5808	5824	5880	5896	6222	6258	6326	6391	6458						
		6523	6588	6655	6791	6849	6902	6949	6995	7055	7088	7127	7155	7178						









CVDMAA.P11 12-DEC-80 15:59

CROSS REFERENCE TABLE -- USER SYMBOLS

EM4	014500	2274	2317	2380	3494#	8283								
EM47A	015675	3494#	4663											
EM47B	015735	3494#	4705											
EM48A	016016	3494#	4958	5003										
EM5	014515	3494#	4155	6224	6260	6328	6393	6460	6525	6590	6657			
EM50A	016067	3494#	6793	7495	8396	8549	8605	8886						
EM50B	016135	3494#	6851	7539										
EM50C	016203	3494#	7255	7986	8485	8757								
EM50D	016251	3494#	6951	7663										
EM50E	016305	3494#	6997	7707										
EM50F	016341	3494#	7057	7775	8421	8630								
EM50G	016406	3494#	7129	7876	8446	8676								
EM50H	016450	3494#	7157	7905										
EM50I	016536	3494#	7180	7929										
EM50J	016600	3494#	7199	7948										
EM50K	016666	3494#	7213	7962										
EM50L	016730	3494#	7297	8038	8696									
EM50M	017012	3494#	7343	8087	8524	8799								
EM50N	017054	3494#	7407	8133										
EM50S	017122	3494#	7826											
EM50U	017174	3494#	8056	8649										
EM50V	017240	3494#	7637	8200										
EM50W	017305	3494#	6904	7090										
EM50X	017354	3494#	8499	8774										
EM50Y	017417	3494#	8859											
EM50Z	017471	3494#	8715											
EM6	014540	3494#	4014											
EM7	014564	3494#	4087	4312	4384									
EM8	014611	3494#	4232											
EM9	014634	3494#	4219											
ENDEMB	012114	3483	3494#											
ENDT7	022614	4272	4372	4386#										
ERRBLK	002244 G	1686#	2195*	2235*	2275*	2318*	2381*	2566*	4664*	4706*	8284*			
ERRFLG	002332	1731#	5205*	5212*	5215									
ERRMSG	002242 G	1685#	2194*	2234*	2274*	2317*	2380*	2565*	4663*	4705*	8283*			
ERRNBR	002240 G	1684#	2193*	2233*	2273*	2316*	2379*	2564*	4662*	4704*	8282*			
ERRTYP	002236 G	1683#	2192*	2232*	2272*	2315*	2378*	2563*	4661*	4703*	8281*			
ERR1	005276 G	2648#	6905											
ERR2	005304 G	2657#	3910	4015	4088	4156	4220	4233	4335	4497	4514	4530		
ERR3	005414 G	2195	2590	2624	2688#									
ERR4	005426 G	2235	2275	2318	2381	2698#	8284							
ERR4\$	011322	2679	2689	3349#										
ERR47	006724 G	3004#	4664	4706										
ERR47.	007110	3054	3062#	4667	4709									
ERR48	007632 G	3106#	4959	5004										
ERR48.	010030	3157	3165#	4962	5008									
ERR5	005552 G	2738#	4313	4385	4438									
ERR5\$	011710	2730	2757	3440#										
ERR50	010762 G	3236#	6794	6852	6952	6998	7058	7091	7130	7158	7181	7200	7214	7256
		7298	7344	7408	7496	7540	7638	7664	7708	7776	7827	7877	7906	7930
		7949	7963	7987	8039	8057	8088	8134	8201	8397	8422	8447	8486	8500
		8525	8550	8606	8631	8650	8677	8697	8716	8758	8775	8800	8860	8887
ERR6	005650 G	2765#	5222											
ERR7	006612 G	2566	2972#	5259	5280	5312	5333	5645	5666	5689	5715	5732	5755	5811
		5827	5883	5899	6225	6261	6329	6394	6461	6526	6591	6658		
ER47CT	007104	3053*	3059#	3064	3066*	3084	4604*							





CVDMAA.P11 12-DEC-80 15:59

CROSS REFERENCE TABLE -- USER SYMBOLS

6618	6634	6642	6661	6716	6718	6727	6736	6741	6751	6760	6769	6778
6787	6812	6817	6827	6836	6845	6854	6866	6876	6886	6897	6913	6925
6936	6945	6961	6973	6982	6991	7000	7012	7022	7032	7042	7051	7060
7065	7075	7084	7099	7104	7114	7123	7140	7151	7164	7174	7191	7206
7225	7230	7240	7249	7267	7272	7282	7291	7300	7313	7318	7328	7337
7346	7359	7364	7374	7383	7392	7401	7411	7418	7425	7434	7442	7453
7462	7471	7480	7489	7508	7513	7524	7533	7542	7556	7565	7574	7585
7595	7604	7613	7622	7631	7644	7655	7670	7680	7692	7701	7710	7722
7732	7742	7751	7760	7769	7778	7783	7793	7802	7811	7820	7836	7841
7852	7861	7870	7887	7899	7912	7923	7940	7955	7974	7979	7999	8008
8013	8023	8032	8041	8049	8066	8071	8081	8090	8103	8108	8118	8127
8143	8148	8158	8167	8176	8185	8194	8203	8206	8366	8368	8375	8384
8389	8399	8413	8424	8434	8439	8453	8461	8473	8478	8491	8502	8512
8517	8527	8537	8542	8553	8557	8564	8574	8584	8593	8598	8608	8622
8633	8642	8652	8664	8669	8679	8688	8699	8708	8725	8733	8745	8750
8760	8766	8777	8787	8792	8802	8807	8817	8826	8835	8844	8853	8874
8879	8889	8899	8902	8920	8960	8976						
1242#	3707	3715										
1242#	3725	3730										
1242#	1244	2601	2635	2653	2684	2694	2734	2761	2968	3000	3058	3161
3323	3637	3697	3717	3732	3745	3767	3807	3809	3851	3875	3912	3914
3930	4017	4019	4037	4100	4102	4114	4159	4161	4174	4236	4238	4264
4266	4315	4317	4319	4337	4387	4389	4390	4392	4411	4440	4499	4516
4532	4538	4540	4559	4565	4580	4615	4617	4861	4867	4894	4913	4942
4974	4987	5105	5107	5139	5224	5261	5282	5314	5335	5339	5341	5357
5363	5371	5382	5386	5388	5393	5395	5412	5418	5426	5437	5441	5443
5448	5450	5468	5474	5484	5492	5503	5507	5509	5514	5516	5581	5590
5603	5613	5762	5768	5770	5773	5834	5840	5842	5845	5906	5912	5914
5916	5918	5935	5941	5949	5960	5964	5966	5971	5973	5990	5996	6004
6015	6019	6021	6026	6028	6045	6051	6059	6070	6074	6076	6080	6082
6105	6111	6120	6131	6135	6137	6142	6144	6161	6168	6183	6194	6202
6214	6239	6250	6264	6266	6284	6291	6305	6313	6332	6334	6349	6356
6370	6378	6397	6399	6414	6421	6438	6446	6464	6466	6481	6488	6502
6510	6529	6531	6546	6553	6567	6575	6594	6596	6611	6618	6634	6642
6661	6663	6716	6718	6727	6736	6741	6751	6760	6769	6778	6787	6812
6817	6827	6836	6845	6854	6866	6876	6886	6897	6913	6925	6936	6945
6961	6973	6982	6991	7000	7012	7022	7032	7042	7051	7060	7065	7075
7084	7099	7104	7114	7123	7140	7151	7164	7174	7191	7206	7225	7230
7240	7249	7267	7272	7282	7291	7300	7313	7318	7328	7337	7346	7359
7364	7374	7383	7392	7401	7411	7413	7418	7425	7434	7442	7453	7462
7471	7480	7489	7508	7513	7524	7533	7542	7556	7565	7574	7585	7595
7604	7613	7622	7631	7644	7655	7670	7680	7692	7701	7710	7722	7732
7742	7751	7760	7769	7778	7783	7793	7802	7811	7820	7836	7841	7852
7861	7870	7887	7899	7912	7923	7940	7955	7974	7979	7999	8008	8013
8023	8032	8041	8049	8066	8071	8081	8090	8103	8108	8118	8127	8143
8148	8158	8167	8176	8185	8194	8203	8205	8206	8208	8366	8368	8375
8384	8389	8399	8413	8424	8434	8439	8453	8461	8473	8478	8491	8502
8512	8517	8527	8537	8542	8553	8555	8557	8564	8574	8584	8593	8598
8608	8622	8633	8642	8652	8664	8669	8679	8688	8699	8708	8725	8733
8745	8750	8760	8766	8777	8787	8792	8802	8807	8817	8826	8835	8844
8853	8874	8879	8889	8899	8901	8902	8904	8943	8965	8976		
1242#	8920	8941										
1242#	1406	1428										
1242#	3516	3635										
1242#												
1242#	1244	8976										

F\$CLEA= 000007  
 F\$DU = 000016  
 F\$END = 000041

F\$HARD= 000004  
 F\$HW = 000013  
 F\$INIT= 000006  
 F\$JMP = 000050  
 F\$MOD = 000000





CVDMAA.P11 12-DEC-80 15:59

CROSS REFERENCE TABLE -- USER SYMBOLS

IENBA = 000001	1587#																			
IENBB = 000020	1588#																			
IENR = 120016	1655#	2483	6125	6625	6647	6652														
IER = 020000 G	1516#																			
IFR = 120015	1654#	6967	7589	8152	8238	8811														
IFRCA1= 000002	1666#																			
IFRCA2= 000001	1667#																			
IFRCB1= 000020	1663#																			
IFRCB2= 000010	1664#																			
IFRIRQ= 000200	1660#																			
IFRSR = 000004	1665#																			
IFRT1 = 000100	1661#	8273																		
IFRT2 = 000040	1662#																			
IHLNK 005222	2594	2596	2602#	3622*																
IHLNK 005274	2628	2630	2636#	3630*																
INITT1 004660	2462#	5596	6729	7427	7558	8377	8586	8891												
INTFLG 002326	1727#	2593*	2627*																	
INTWCH 002330	1729#	2582	2616	3631*																
IRQA = 000004	1563#																			
IRQB = 000002	1564#																			
IRQREG= 123005	1562#																			
ISR = 000100 G	1509#																			
IXE = 004000 G	1514#																			
ISAU = 000041	1242#	3741#	3745#																	
ISAUTO= 000041	1242#	3662#	3697#																	
ISCLN = 000041	1242#	3707#	3717#																	
ISDU = 000041	1242#	3725#	3732#																	
ISHRD = 000041	8920#	8943#																		
ISINIT= 000041	1242#	3516#	3637#																	
ISMOD = 000041	1242#	1244#	8976#																	
ISMSG = 000041	1242#	2648#	2653#	2657#	2684#	2688#	2694#	2698#	2734#	2738#	2761#	2765#	2968#							
	2972#	3000#	3004#	3058#	3106#	3161#	3236#	3323#	3820#	3851#										
ISPROT= 000040	1242#	3503#																		
ISPTAB= 000041	1242#																			
ISPR = 000041	1242#																			
ISRPT = 000041	1242#																			
ISSEG = 000041	1242#	3767	3875	3930	4037	4114	4174	4264	4266	4319	4411	4559	4861							
	5139	5357	5371	5412	5426	5468	5492	5581	5613	5623#	5762#	5773	5790#							
	5834#	5845	5862#	5906#	5935	5949	5990	6004	6045	6059	6105	6120	6161							
	6284	6349	6414	6481	6546	6611	6716	6718	7418	8366	8368	8557								
ISSETU= 000041	1242#																			
ISSFT = 000041	8960#	8965#																		
ISSRV = 000041	1242#	2580#	2601#	2614#	2635#															
ISSUB = 000041	1242#	3767	3875	3930	4037	4114	4174	4264	4266#	4315#	4317#	4319#	4387#							
	4389#	4411	4559	4861	5139	5357	5371#	5386#	5388#	5412	5426#	5441#	5443#							
	5468	5492#	5507#	5509#	5581	5613#	5768#	5770#	5773#	5840#	5842#	5845#	5912#							
	5914#	5935	5949#	5964#	5966#	5990	6004#	6019#	6021#	6045	6059#	6074#	6076#							
	6105	6120#	6135#	6137#	6161	6284	6349	6414	6481	6546	6611	6716	6718#							
	6741	6817	6854	6897	7000	7032	7060	7065	7104	7164	7206	7230	7272							
	7300	7318	7346	7364	7411#	7413#	7418#	7442	7513	7542	7585	7710	7732							
	7778	7783	7841	7912	7955	7979	8013	8041	8049	8071	8090	8108	8148							
	8203#	8205#	8366	8368#	8389	8399	8413	8424	8439	8461	8478	8491	8502							
	8517	8527	8542	8553#	8555#	8557#	8598	8608	8622	8633	8642	8652	8669							
	8679	8688	8699	8708	8733	8750	8760	8766	8777	8792	8802	8807	8879							
ISTST = 000041	1242#	3767#	3807#	3809#	3875#	3912#	3914#	3930#	4017#	4019#	4037#	4100#	4102#							

CVDMAA.P11 12-DEC-80 15:59

CROSS REFERENCE TABLE -- USER SYMBOLS

4114#	4159#	4161#	4174#	4236#	4238#	4264#	4266	4319	4337	4390#	4392#	4411#
4440	4499	4516	4532	4538#	4540#	4559#	4565	4580	4615#	4617#	4861#	4867
4894	4913	4942	4974	4987	5105#	5107#	5139#	5224	5261	5282	5314	5335
5339#	5341#	5357#	5363	5371	5382	5393#	5395#	5412#	5418	5426	5437	5448#
5450#	5468#	5474	5484	5492	5503	5514#	5516#	5581#	5590	5603	5613	5773
5845	5916#	5918#	5935#	5941	5949	5960	5971#	5973#	5990#	5996	6004	6015
6026#	6028#	6045#	6051	6059	6070	6080#	6082#	6105#	6111	6120	6131	6142#
6144#	6161#	6168	6183	6194	6202	6214	6239	6250	6264#	6266#	6284#	6291
6305	6313	6332#	6334#	6349#	6356	6370	6378	6397#	6399#	6414#	6421	6438
6446	6464#	6466#	6481#	6488	6502	6510	6529#	6531#	6546#	6553	6567	6575
6594#	6596#	6611#	6618	6634	6642	6661#	6663#	6716#	6718	6727	6736	6751
6760	6769	6778	6787	6812	6827	6836	6845	6866	6876	6886	6913	6925
6936	6945	6961	6973	6982	6991	7012	7022	7042	7051	7075	7084	7099
7114	7123	7140	7151	7174	7191	7225	7240	7249	7267	7282	7291	7313
7328	7337	7359	7374	7383	7392	7401	7418	7425	7434	7453	7462	7471
7480	7489	7508	7524	7533	7556	7565	7574	7595	7604	7613	7622	7631
7644	7655	7670	7680	7692	7701	7722	7742	7751	7760	7769	7793	7802
7811	7820	7836	7852	7861	7870	7887	7899	7923	7940	7974	7999	8008
8023	8032	8066	8081	8103	8118	8127	8143	8158	8167	8176	8185	8194
8206#	8208#	8366#	8368	8375	8384	8434	8453	8473	8512	8537	8557	8564
8574	8584	8593	8664	8725	8745	8787	8817	8826	8835	8844	8853	8874
8902#	8904#											
1242#												
6906	7639	7665	8264#	8448	8720							
6888	7024	7576	7724	8221#	8404	8613						
1517#												
1725#	3571*	3577*	3579	3691								
1506#												
1550#												
1547#												
1244#												
1346#												
1304#												
1331	3741#											
1330#												
1347	3662#											
1344#												
1345	3707#											
1300#												
1282#												
1337	2121#											
1336#												
1322#												
1307	1367#											
1352#												
1306#												
1302#												
1333	3725#											
1332#												
1323	2109#											
1317#												
1310#												
1341	1682#											
1340#												
1312#												
1326#												

JSJMP = 000167  
 KICKT1 036150  
 LODT1C 036060  
 LOE = 040000 G  
 LOGDEV 002322  
 LOT = 000010 G  
 LSIDCL= 000002  
 LSIHLT= 000020  
 LUIMOD 002000 G  
 LSACP 002110 G  
 LSAPT 002036 G  
 LSAU 020352 G  
 LSAUT 002070 G  
 LSAUTO 020204 G  
 LSCCP 002106 G  
 LSCLEA 020330 G  
 LSCO 002032 G  
 LSDEPO 002011 G  
 LSDESC 003542 G  
 LSDESP 002076 G  
 LSDEVP 002060 G  
 LSDISP 002124 G  
 LSDLY 002116 G  
 LSDTP 002040 G  
 LSDTYP 002034 G  
 LSDU 020346 G  
 LSDUT 002072 G  
 LSDVTY 003522 G  
 LSEF 002052 G  
 LSENV1 002044 G  
 LSERRT 002236 G  
 LSETP 002102 G  
 LSEXP1 002046 G  
 LSEXP4 002064 G



CVDMAA.P11 12-DEC-80 15:59

## CROSS REFERENCE TABLE -- USER SYMBOLS

LSEXP5	002066	G	1328#		
LSHARD	040066	G	1289	8920	8921#
LSHIME	002120	G	1354#		
LSHPCP	002016	G	1283#		
LSHPTP	002022	G	1292#		
LSHW	002216	G	1293	1406	1407#
LSICP	002104	G	1342#		
LSINIT	017622	G	1343	3516#	
LSLADP	002026	G	1296#		
LSLAST	040344	G	1297	8980#	
LSLOAD	002100	G	1338#		
LSLUN	002074	G	1334#		
LSMREV	002050	G	1314#		
LSNAME	002000	G	1271#		
LSPRIO	002042	G	1308#		
LSPROT	017614	G	1349	3503#	
LSPRT	002112	G	1348#		
LSREPP	002062	G	1324#		
LSREV	002010	G	1280#		
LSSOFT	040232	G	8960	8961#	
LSSPC	002056	G	1320#		
LSSPCP	002020	G	1290#		
LSSPTP	002024	G	1294#		
LSSTA	002030	G	1298#		
LSSW	002236	G	1437	1438#	
LSTEST	002114	G	1350#		
LSTIML	002014	G	1286#		
LSUNIT	002012	G	1284#		
L10000	002234		1406	1428#	
L10001	002236		1437	1441#	
L10002	005220		2599#		
L10003	005272		2633#		
L10004	005302		2651#		
L10005	005412		2682#		
L10006	005424		2692#		
L10007	005550		2732#		
L10010	005646		2759#		
L10011	006610		2966#		
L10012	006722		2998#		
L10013	007102		3056#		
L10014	010022		3159#		
L10015	011274		3321#		
L10017	020172		3635#		
L10020	020320		3695#		
L10021	020344		3715#		
L10022	020350		3730#		
L10023	020352		3743#		
L10024	020510		3807#		
L10025	020640		3849#		
L10026	021226		3912#		
L10027	021420		4017#		
L10030	021562		4100#		
L10031	021704		4159#		
L10032	022106		4236#		
L10033	022616		4338	4390#	
L10034	022326		4315#		







CVDMAA.P11 12-DEC-80 15:59

CROSS REFERENCE TABLE -- USER SYMBOLS

PCR = 120014	1653#	6064	6558	6580	6585									
PFLAG 002350	1738#	3115	3140	3197	4175									
PNT = 001000 G	1512#													
PRI = 002000 G	1513#													
PRIPTY 040177	8935	8944#												
PRI00 = 000000 G	1501#													
PRI01 = 000040 G	1500#													
PRI02 = 000100 G	1499#													
PRI03 = 000140 G	1498#													
PRI04 = 000200 G	1497#													
PRI05 = 000240 G	1496#													
PRI06 = 000300 G	1495#													
PRI07 = 000340 G	1494#													
PSTACK 002324	1726#	3518*												
PU24 = 000001	1569#													
RGRAM 023570	4593	4675#												
READ 004064	2257#	2487	2550	4649	4691	4935	4980	5186	5243	5265	5300	5317	5634	
	5649	5654	5678	5704	5719	5744	5800	5815	5872	5887	6207	6243	6317	
	6382	6449	6514	6579	6646	6744	6753	6762	6771	6780	6820	6829	6838	
	6859	6918	6929	6938	6954	6966	6975	6984	7035	7044	7068	7077	7092	
	7107	7116	7133	7144	7167	7184	7218	7233	7242	7275	7284	7321	7330	
	7367	7376	7385	7394	7446	7455	7464	7473	7482	7517	7526	7567	7588	
	7597	7606	7615	7624	7648	7673	7685	7694	7735	7744	7753	7762	7786	
	7795	7804	7813	7829	7845	7854	7863	7880	7892	7916	7933	7967	8016	
	8025	8074	8111	8120	8151	8160	8169	8178	8187	8237	8265	8301	8427	
	8466	8657	8738	8810	8819	8828	8837	8846						
READI 004176	2299#	5229												
REDLOC= 000001	1606#	2258	2269	2301	2312									
REDPAG= 000003	1608#													
REGNUM 002334	1732#	2559*	2560*	2659	2740	2973	3904*	4009*	4080*	4150*	4214*	4227*	4282*	
	4289*	4296*	4303*	4329*	4350*	4359*	4368*	4375*	4432*	4491*	4508*	4524*	5253*	
	5274*	5306*	5327*	5639*	5660*	5683*	5709*	5722*	5749*	5805*	5821*	5877*	5893*	
	6219*	6255*	6323*	6388*	6455*	6520*	6585*	6652*						
REG0 002420	1814#													
REG1 002422	1815#													
REG2 002424	1816#													
REG3 002426	1817#													
REG4 002430	1818#													
REG5 002432	1819#													
REG6 002434	1820#													
REG7 002436	1821#													
RESFMC 003040	2048#	3888	3890	3902										
RESFT3 003062	2048	2068#	3951	3987*	3994	4005								
RESTRY 017740	3530	3564#												
RUN = 000200	1596#	2165	2175	2219	2229	4072	4446	4472						
SEL0 002352	1784#	2219*	2425	4277	4280	4430								
SEL10 002372	1796#	2429												
SEL12 002376	1799#	2430												
SEL14 002402	1802#	2431												
SEL16 002406	1805#	2432												
SEL2 002356	1787#	2426	4285	4287	4479	4487								
SEL4 002362	1790#	2257*	2300*	2342*	2363*	2427	4274*	4292	4294	8270*				
SEL6 002366	1793#	2323	2343*	2364*	2428	4275*	4299	4301	8271*					
SFPTBL 002236 G	1439#													
SLTO = 000020	1624#	1625	1626	1627	1628	1629	1630	1631	1632	1633	1634	1635	4274	
	4292	4295	4363	4366	4422	4632	4634	4680	4682	4885	5027	5066		



CVDMAA.P11 12-DEC-80 15:59

CROSS REFERENCE TABLE -- USER SYMBOLS

SLT2 = 000022	1627#													
SLT4 = 000024	1630#													
SLT6 = 000026	1633#													
SR = 120012	1651#	5954	6429	6450	6455									
STALL 005032	2520#	7646	7647											
STARST 017674	3524	3547#												
STREG 005034	2542#	5377	5432	5498	5955	6010	6065	6126	6178	6189	6234	6300	6365	
	6433	6497	6562	6629										
SVCGBL= 000000	1242#	1244	1251#	1271	1280	1282	1284	1286	1288	1290	1292	1294	1296	
	1298	1300	1302	1304	1306	1308	1310	1312	1314	1317	1320	1322	1324	
	1326	1328	1330	1332	1334	1336	1338	1340	1342	1344	1346	1348	1350	
	1352	1354	1367	1407	1408	1438	1439	1682	2109	2121	2580	2614	2648	
	2657	2688	2698	2738	2765	2972	3004	3106	3236	3503	3516	3662	3707	
	3725	3741	3820	8921	8961	8980#	8981							
SVCINS= 000001	1242#	1248#	1272	1273	1274	1275	1276	1277	1278	1279	1281	1283	1285	
	1287	1289	1291	1293	1295	1297	1299	1301	1303	1305	1307	1309	1311	
	1313	1315	1316	1318	1319	1321	1323	1325	1327	1329	1331	1333	1335	
	1337	1339	1341	1343	1345	1347	1349	1351	1353	1355	1366	1368	1369	
	1370	1371	1372	1373	1374	1375	1376	1377	1378	1379	1380	1381	1382	
	1383	1384	1385	1386	1387	1388	1389	1390	1391	1392	1393	1394	1395	
	1406	1437	2110	2113	2122	2129	2192	2193	2194	2195	2232	2233	2234	
	2235	2272	2273	2274	2275	2315	2316	2317	2318	2378	2379	2380	2381	
	2563	2564	2565	2566	2587	2588	2589	2590	2600	2621	2622	2623	2624	
	2634	2652	2659	2660	2661	2662	2663	2664	2665	2668	2669	2670	2671	
	2672	2673	2674	2675	2676	2677	2678	2683	2693	2704	2705	2706	2707	
	2708	2709	2710	2720	2721	2722	2723	2724	2725	2726	2727	2733	2740	
	2741	2742	2743	2744	2745	2746	2749	2750	2751	2752	2753	2754	2755	
	2756	2760	2770	2771	2772	2773	2774	2775	2777	2778	2779	2780	2781	
	2782	2783	2784	2785	2786	2787	2788	2789	2790	2791	2792	2793	2794	
	2796	2797	2798	2799	2800	2801	2802	2803	2804	2807	2808	2809	2810	
	2811	2812	2813	2814	2815	2816	2817	2818	2819	2820	2821	2822	2823	
	2824	2826	2827	2828	2829	2830	2831	2832	2833	2834	2837	2838	2839	
	2840	2841	2842	2843	2844	2845	2846	2847	2848	2849	2850	2851	2852	
	2853	2854	2856	2857	2858	2859	2860	2861	2862	2863	2864	2868	2869	
	2870	2871	2872	2873	2875	2876	2877	2878	2879	2880	2881	2882	2883	
	2884	2885	2886	2887	2888	2889	2890	2891	2892	2894	2895	2896	2897	
	2898	2899	2900	2901	2902	2905	2906	2907	2908	2909	2910	2911	2912	
	2913	2914	2915	2916	2917	2918	2919	2920	2921	2922	2924	2925	2926	
	2927	2928	2929	2930	2931	2932	2935	2936	2937	2938	2939	2940	2941	
	2942	2943	2944	2945	2946	2947	2948	2949	2950	2951	2952	2954	2955	
	2956	2957	2958	2959	2960	2961	2962	2967	2976	2977	2978	2979	2980	
	2981	2982	2985	2986	2987	2988	2989	2990	2991	2992	2993	2994	2995	
	2999	3015	3016	3017	3018	3019	3020	3024	3025	3026	3027	3028	3032	
	3033	3034	3035	3036	3040	3041	3042	3043	3044	3048	3049	3050	3051	
	3052	3057	3072	3073	3074	3075	3076	3077	3078	3079	3080	3081	3082	
	3083	3087	3088	3089	3090	3091	3110	3111	3112	3113	3114	3118	3119	
	3120	3121	3122	3126	3127	3128	3129	3130	3135	3136	3137	3138	3139	
	3143	3144	3145	3146	3147	3151	3152	3153	3154	3155	3160	3184	3185	
	3186	3187	3188	3189	3190	3191	3192	3193	3194	3195	3196	3208	3209	
	3210	3211	3212	3213	3214	3215	3216	3217	3221	3222	3223	3224	3225	
	3249	3250	3251	3252	3253	3254	3258	3259	3260	3261	3262	3266	3267	
	3268	3269	3270	3271	3272	3273	3274	3275	3276	3277	3278	3279	3281	
	3282	3283	3284	3285	3286	3287	3288	3289	3293	3294	3295	3296	3297	
	3298	3299	3300	3301	3302	3303	3304	3305	3306	3308	3309	3310	3311	
	3312	3313	3314	3315	3316	3322	3350	3351	3352	3353	3354	3355	3356	
	3358	3359	3360	3361	3362	3363	3364	3365	3366	3367	3368	3369	3370	



CROSS REFERENCE TABLE -- USER SYMBOLS

3372	3373	3374	3375	3376	3377	3379	3380	3381	3382	3383	3384	3385
3386	3387	3388	3389	3390	3391	3393	3394	3395	3396	3397	3398	3400
3401	3402	3403	3404	3405	3406	3407	3408	3409	3410	3411	3412	3414
3415	3416	3417	3418	3419	3421	3422	3423	3424	3425	3426	3427	3428
3429	3430	3431	3432	3433	3442	3443	3444	3445	3446	3447	3448	3450
3451	3452	3453	3454	3455	3456	3457	3458	3460	3461	3462	3463	3464
3465	3467	3468	3469	3470	3471	3472	3473	3474	3475	3483	3484	3485
3486	3487	3521	3522	3524	3527	3528	3530	3533	3534	3536	3539	3540
3542	3553	3554	3555	3556	3557	3558	3561	3562	3579	3580	3581	3583
3616	3617	3618	3619	3620	3621	3624	3625	3626	3627	3628	3629	3636
3664	3665	3666	3667	3668	3669	3686	3687	3691	3692	3696	3709	3710
3712	3713	3716	3728	3731	3744	3772	3773	3774	3775	3776	3777	3794
3795	3802	3803	3804	3805	3808	3822	3823	3824	3825	3826	3827	3829
3830	3831	3832	3833	3835	3836	3837	3838	3839	3841	3842	3843	3844
3845	3846	3847	3850	3882	3907	3908	3909	3910	3913	4012	4013	4014
4015	4018	4044	4085	4086	4087	4088	4101	4123	4139	4153	4154	4155
4156	4160	4186	4194	4217	4218	4219	4220	4230	4231	4232	4233	4237
4267	4271	4310	4311	4312	4313	4316	4320	4332	4333	4334	4335	4337
4338	4382	4383	4384	4385	4388	4391	4416	4426	4435	4436	4437	4438
4440	4441	4494	4495	4496	4497	4499	4500	4511	4512	4513	4514	4516
4517	4527	4528	4529	4530	4532	4533	4539	4563	4565	4566	4578	4580
4581	4586	4596	4601	4613	4616	4661	4662	4663	4664	4703	4704	4705
4706	4865	4867	4868	4892	4894	4895	4911	4913	4914	4933	4940	4942
4943	4956	4957	4958	4959	4972	4974	4975	4985	4987	4988	5001	5002
5003	5004	5103	5106	5143	5151	5158	5176	5191	5219	5220	5221	5222
5224	5225	5234	5241	5248	5256	5257	5258	5259	5261	5262	5270	5277
5278	5279	5280	5282	5283	5290	5297	5309	5310	5311	5312	5314	5315
5322	5330	5331	5332	5333	5335	5336	5340	5361	5363	5364	5372	5380
5382	5383	5387	5394	5416	5418	5419	5427	5435	5437	5438	5442	5449
5472	5474	5475	5482	5484	5485	5493	5501	5503	5504	5508	5515	5588
5590	5591	5601	5603	5604	5614	5623	5642	5643	5644	5645	5663	5664
5665	5666	5686	5687	5688	5689	5712	5713	5714	5715	5729	5730	5731
5732	5752	5753	5754	5755	5761	5769	5774	5790	5808	5809	5810	5811
5824	5825	5826	5827	5833	5841	5846	5862	5880	5881	5882	5883	5896
5897	5898	5899	5905	5913	5917	5939	5941	5942	5950	5958	5960	5961
5965	5972	5994	5996	5997	6005	6013	6015	6016	6020	6027	6049	6051
6052	6060	6068	6070	6071	6075	6081	6109	6111	6112	6121	6129	6131
6132	6136	6143	6166	6168	6169	6181	6183	6184	6192	6194	6195	6200
6202	6203	6212	6214	6215	6222	6223	6224	6225	6237	6239	6240	6248
6250	6251	6258	6259	6260	6261	6265	6289	6291	6292	6303	6305	6306
6311	6313	6314	6326	6327	6328	6329	6333	6354	6356	6357	6368	6370
6371	6376	6378	6379	6391	6392	6393	6394	6398	6419	6421	6422	6436
6438	6439	6444	6446	6447	6458	6459	6460	6461	6465	6486	6488	6489
6500	6502	6503	6508	6510	6511	6523	6524	6525	6526	6530	6551	6553
6554	6565	6567	6568	6573	6575	6576	6588	6589	6590	6591	6595	6616
6618	6619	6632	6634	6635	6640	6642	6643	6655	6656	6657	6658	6662
6719	6725	6727	6728	6734	6736	6737	6741	6742	6749	6751	6752	6758
6760	6761	6767	6769	6770	6776	6778	6779	6785	6787	6788	6791	6792
6793	6794	6796	6797	6798	6799	6800	6810	6812	6813	6817	6818	6825
6827	6828	6834	6836	6837	6843	6845	6846	6849	6850	6851	6852	6854
6855	6864	6866	6867	6874	6876	6877	6884	6886	6887	6897	6898	6902
6903	6904	6905	6911	6913	6914	6923	6925	6926	6934	6936	6937	6943
6945	6946	6949	6950	6951	6952	6959	6961	6962	6971	6973	6974	6980
6982	6983	6989	6991	6992	6995	6996	6997	6998	7000	7001	7010	7012
7013	7020	7022	7023	7032	7033	7040	7042	7043	7049	7051	7052	7055
7056	7057	7058	7060	7061	7065	7066	7073	7075	7076	7082	7084	7085



CVDMAA.P11

12-DEC-80 15:59

CROSS REFERENCE TABLE -- USER SYMBOLS

7088	7089	7090	7091	7097	7099	7100	7104	7105	7112	7114	7115	7121
7123	7124	7127	7128	7129	7130	7138	7140	7141	7149	7151	7152	7155
7156	7157	7158	7164	7165	7172	7174	7175	7178	7179	7180	7181	7189
7191	7192	7197	7198	7199	7200	7206	7207	7211	7212	7213	7214	7223
7225	7226	7230	7231	7238	7240	7241	7247	7249	7250	7253	7254	7255
7256	7265	7267	7268	7272	7273	7280	7282	7283	7289	7291	7292	7295
7296	7297	7298	7300	7301	7311	7313	7314	7318	7319	7326	7328	7329
7335	7337	7338	7341	7342	7343	7344	7346	7347	7357	7359	7360	7364
7365	7372	7374	7375	7381	7383	7384	7390	7392	7393	7399	7401	7402
7405	7406	7407	7408	7412	7419	7423	7425	7426	7432	7434	7435	7442
7443	7451	7453	7454	7460	7462	7463	7469	7471	7472	7478	7480	7481
7487	7489	7490	7493	7494	7495	7496	7506	7508	7509	7513	7514	7522
7524	7525	7531	7533	7534	7537	7538	7539	7540	7542	7543	7554	7556
7557	7563	7565	7566	7572	7574	7575	7585	7586	7593	7595	7596	7602
7604	7605	7611	7613	7614	7620	7622	7623	7629	7631	7632	7635	7636
7637	7638	7642	7644	7645	7653	7655	7656	7661	7662	7663	7664	7668
7670	7671	7678	7680	7681	7690	7692	7693	7699	7701	7702	7705	7706
7707	7708	7710	7711	7720	7722	7723	7732	7733	7740	7742	7743	7749
7751	7752	7758	7760	7761	7767	7769	7770	7773	7774	7775	7776	7778
7779	7783	7784	7791	7793	7794	7800	7802	7803	7809	7811	7812	7818
7820	7821	7824	7825	7826	7827	7834	7836	7837	7841	7842	7850	7852
7853	7859	7861	7862	7868	7870	7871	7874	7875	7876	7877	7885	7887
7888	7897	7899	7900	7903	7904	7905	7906	7912	7913	7921	7923	7924
7927	7928	7929	7930	7938	7940	7941	7946	7947	7948	7949	7955	7956
7960	7961	7962	7963	7972	7974	7975	7979	7980	7984	7985	7986	7987
7997	7999	8000	8006	8008	8009	8013	8014	8021	8023	8024	8030	8032
8033	8036	8037	8038	8039	8041	8042	8049	8050	8054	8055	8056	8057
8064	8066	8067	8071	8072	8079	8081	8082	8085	8086	8087	8088	8090
8091	8101	8103	8104	8108	8109	8116	8118	8119	8125	8127	8128	8131
8132	8133	8134	8141	8143	8144	8148	8149	8156	8158	8159	8165	8167
8168	8174	8176	8177	8183	8185	8186	8192	8194	8195	8198	8199	8200
8201	8204	8207	8242	8281	8282	8283	8284	8306	8369	8373	8375	8376
8382	8384	8385	8389	8390	8394	8395	8396	8397	8399	8400	8413	8414
8419	8420	8421	8422	8424	8425	8432	8434	8435	8439	8440	8444	8445
8446	8447	8451	8453	8454	8461	8462	8471	8473	8474	8478	8479	8483
8484	8485	8486	8491	8492	8497	8498	8499	8500	8502	8503	8510	8512
8513	8517	8518	8522	8523	8524	8525	8527	8528	8535	8537	8538	8542
8543	8547	8548	8549	8550	8554	8558	8562	8564	8565	8572	8574	8575
8582	8584	8585	8591	8593	8594	8598	8599	8603	8604	8605	8606	8608
8609	8622	8623	8628	8629	8630	8631	8633	8634	8642	8643	8647	8648
8649	8650	8652	8653	8662	8664	8665	8669	8670	8674	8675	8676	8677
8679	8680	8688	8689	8694	8695	8696	8697	8699	8700	8708	8709	8713
8714	8715	8716	8723	8725	8726	8733	8734	8743	8745	8746	8750	8751
8755	8756	8757	8758	8760	8761	8766	8767	8772	8773	8774	8775	8777
8778	8785	8787	8788	8792	8793	8797	8798	8799	8800	8802	8803	8807
8808	8815	8817	8818	8824	8826	8827	8833	8835	8836	8842	8844	8845
8851	8853	8854	8857	8858	8859	8860	8872	8874	8875	8879	8880	8884
8885	8886	8887	8889	8890	8896	8900	8903	8920	8924	8925	8926	8927
8929	8930	8931	8932	8934	8935	8936	8937	8938	8941	8960	8963	8977
8978	8979											
1242#	1250#	4266	4319	5371	5426	5492	5613	5773	5845	5949	6004	6059
6120	6718	7418	8368	8557								
1242#	1252#	1428	1441	2599	2633	2651	2682	2692	2732	2759	2966	2998
3056	3159	3321	3635	3695	3715	3730	3743	3807	3849	3912	4017	4100
4159	4236	4315	4387	4390	4538	4615	5105	5339	5386	5393	5441	5448
5507	5514	5760	5768	5832	5840	5904	5912	5916	5964	5971	6019	6026

SVCSUB= 000001

SVCTAG= 000001











CVDMAA.P11 12-DEC-80 15:59

CROSS REFERENCE TABLE -- USER SYMBOLS

TSCODE= 002032  
TSERRN= 000146

8924#	8929#	8934#											
1242#	2588#	2622#	3803#	3908#	4013#	4086#	4154#	4218#	4231#	4311#	4333#	4383#	
4436#	4495#	4512#	4528#	4957#	5002#	5220#	5257#	5278#	5310#	5331#	5643#	5664#	
5687#	5713#	5730#	5753#	5809#	5825#	5881#	5897#	6223#	6259#	6327#	6392#	6459#	
6524#	6589#	6656#	6792#	6850#	6903#	6950#	6996#	7056#	7089#	7128#	7156#	7179#	
7198#	7212#	7254#	7296#	7342#	7406#	7494#	7538#	7636#	7662#	7706#	7774#	7825#	
7875#	7904#	7928#	7947#	7961#	7985#	8037#	8055#	8086#	8132#	8199#	8395#	8420#	
8445#	8484#	8498#	8523#	8548#	8604#	8629#	8648#	8675#	8695#	8714#	8756#	8773#	
8798#	8858#	8885#											

TSEXCP= 000000  
TSFLAG= 000040

8924#	8928	8929#	8933	8934#	8939								
4337#	4440#	4499#	4516#	4532#	4565#	4580#	4867#	4894#	4913#	4942#	4974#	4987#	
5224#	5261#	5282#	5314#	5335#	5363#	5382#	5418#	5437#	5474#	5484#	5503#	5590#	
5603#	5941#	5960#	5996#	6015#	6051#	6070#	6111#	6131#	6168#	6183#	6194#	6202#	
6214#	6239#	6250#	6291#	6305#	6313#	6356#	6370#	6378#	6421#	6438#	6446#	6488#	
6502#	6510#	6553#	6567#	6575#	6618#	6634#	6642#	6727#	6736#	6741#	6751#	6760#	
6769#	6778#	6787#	6812#	6817#	6827#	6836#	6845#	6854#	6866#	6876#	6886#	6897#	
6913#	6925#	6936#	6945#	6961#	6973#	6982#	6991#	7000#	7012#	7022#	7032#	7042#	
7051#	7060#	7065#	7075#	7084#	7099#	7104#	7114#	7123#	7140#	7151#	7164#	7174#	
7191#	7206#	7225#	7230#	7240#	7249#	7267#	7272#	7282#	7291#	7300#	7313#	7318#	
7328#	7337#	7346#	7359#	7364#	7374#	7383#	7392#	7401#	7425#	7434#	7442#	7453#	
7462#	7471#	7480#	7489#	7508#	7513#	7524#	7533#	7542#	7556#	7565#	7574#	7585#	
7595#	7604#	7613#	7622#	7631#	7644#	7655#	7670#	7680#	7692#	7701#	7710#	7722#	
7732#	7742#	7751#	7760#	7769#	7778#	7783#	7793#	7802#	7811#	7820#	7836#	7841#	
7852#	7861#	7870#	7887#	7899#	7912#	7923#	7940#	7955#	7974#	7979#	7999#	8008#	
8013#	8023#	8032#	8041#	8049#	8066#	8071#	8081#	8090#	8103#	8108#	8118#	8127#	
8143#	8148#	8158#	8167#	8176#	8185#	8194#	8375#	8384#	8389#	8399#	8413#	8424#	
8434#	8439#	8453#	8461#	8473#	8478#	8491#	8502#	8512#	8517#	8527#	8537#	8542#	
8564#	8574#	8584#	8593#	8598#	8608#	8622#	8633#	8642#	8652#	8664#	8669#	8679#	
8688#	8699#	8708#	8725#	8733#	8745#	8750#	8760#	8766#	8777#	8787#	8792#	8802#	
8807#	8817#	8826#	8835#	8844#	8853#	8874#	8879#	8889#					

TSGMAN= 000000  
TSHILI= 000007  
TSLAST= 000001  
TSLOLI= 000000  
TSLSYM= 010000

1242#													
8924#	8927	8929#	8932	8934#	8938								
1242#	8978#												
8924#	8926	8929#	8931	8934#	8937								
1242#	1429	1442	2600	2634	2652	2683	2693	2733	2760	2967	2999	3057	
3160	3322	3636	3696	3716	3731	3744	3808	3850	3913	4018	4101	4160	
4237	4316	4388	4391	4539	4616	5106	5340	5387	5394	5442	5449	5508	
5515	5769	5841	5913	5917	5965	5972	6020	6027	6075	6081	6136	6143	
6265	6333	6398	6465	6530	6595	6662	7412	8204	8207	8554	8900	8903	
8943	8965												

TSLTNO= 000034  
TSNEST= 177777

8981#													
1242#	1244#	1406#	1428#	1437#	1441#	2580#	2599#	2614#	2633#	2648#	2651#	2657#	
2682#	2688#	2692#	2698#	2732#	2738#	2759#	2765#	2966#	2972#	2998#	3004#	3056#	
3106#	3159#	3236#	3321#	3503#	3508#	3516#	3635#	3662#	3695#	3707#	3715#	3725#	
3730#	3741#	3743#	3768#	3807#	3820#	3849#	3876#	3912#	3931#	4017#	4038#	4100#	
4115#	4159#	4175#	4236#	4265#	4267#	4315#	4320#	4387#	4390#	4412#	4538#	4560#	
4615#	4862#	5105#	5140#	5339#	5358#	5372#	5386#	5393#	5413#	5427#	5441#	5448#	
5469#	5493#	5507#	5514#	5582#	5614#	5623#	5760#	5768#	5774#	5790#	5832#	5840#	
5846#	5862#	5904#	5912#	5916#	5936#	5950#	5964#	5971#	5991#	6005#	6019#	6026#	
6046#	6060#	6074#	6080#	6106#	6121#	6135#	6142#	6162#	6264#	6285#	6332#	6350#	
6397#	6415#	6464#	6482#	6529#	6547#	6594#	6612#	6661#	6717#	6719#	7411#	7419#	
8203#	8206#	8367#	8369#	8553#	8558#	8899#	8902#	8920#	8941#	8960#	8963#	8976#	

TSNSO = 000000  
TSNS1 = 000005

1244#	8976												
1406#	1428	1437#	1441	2580#	2599	2614#	2633	2648#	2651	2657#	2682	2688#	
2692	2698#	2732	2738#	2759	2765#	2966	2972#	2998	3004#	3056	3106#	3159	
3236#	3321	3503#	3508	3516#	3635	3662#	3695	3707#	3715	3725#	3730	3741#	



CVDMAA.P11 12-DEC-80 15:59

CROSS REFERENCE TABLE -- USER SYMBOLS

	3743	3768#	3807	3820#	3849	3876#	3912	3931#	4017	4038#	4100	4115#	4159
	4175#	4236	4265#	4390	4412#	4538	4560#	4615	4862#	5105	5140#	5339	5358#
	5393	5413#	5448	5469#	5514	5582#	5916	5936#	5971	5991#	6026	6046#	6080
	6106#	6142	6162#	6264	6285#	6332	6350#	6397	6415#	6464	6482#	6529	6547#
	6594	6612#	6661	6717#	8206	8367#	8902	8920#	8941	8960#	8963		
TSNS2 = 000002	4267#	4315	4320#	4387	5372#	5386	5427#	5441	5493#	5507	5614#	5768	5774#
	5840	5846#	5912	5950#	5964	6005#	6019	6060#	6074	6121#	6135	6719#	7411
	7419#	8203	8369#	8553	8558#	8899							
	5623#	5760	5790#	5832	5862#	5904							
TSNS3 = 000003	1242#												
TSPTNU= 000000	1242#												
TSSAVL= 177777	1242#												
TSSEGL= 177777	1242#	5623#	5760#	5762	5790#	5832#	5834	5862#	5904#	5906			
TSSEKO= 010000	5623#	5760	5790#	5832	5862#	5904							
TSSUBN= 000002	1242#	3767#	3875#	3930#	4037#	4114#	4174#	4264#	4266#	4319#	4411#	4559#	4861#
	5139#	5357#	5371#	5412#	5426#	5468#	5492#	5581#	5613#	5773#	5845#	5935#	5949#
	5990#	6004#	6045#	6059#	6105#	6120#	6161#	6284#	6349#	6414#	6481#	6546#	6611#
	6716#	6718#	7418#	8366#	8368#	8557#							
TSTAGL= 177777	1242#												
TSTAGN= 010103	1242#	1406#	1437#	2580#	2614#	2648#	2657#	2688#	2698#	2738#	2765#	2972#	3004#
	3106#	3236#	3503#	3516#	3662#	3707#	3725#	3741#	3768#	3820#	3876#	3931#	4038#
	4115#	4175#	4265#	4267#	4320#	4412#	4560#	4862#	5140#	5358#	5372#	5413#	5427#
	5469#	5493#	5582#	5614#	5774#	5846#	5936#	5950#	5991#	6005#	6046#	6060#	6106#
	6121#	6162#	6285#	6350#	6415#	6482#	6547#	6612#	6717#	6719#	7419#	8367#	8369#
	8558#	8920#	8960#										
TSTEMP= 000000	1368#	1369#	1370#	1371#	1372#	1373#	1374#	1375#	1376#	1377#	1378#	1379#	1380#
	1381#	1382#	1383#	1384#	1385#	1386#	1387#	1388#	1389#	1390#	1391#	1392#	1393#
	1394#	1395#	1396#	1428#	1441#	2599#	2633#	2651#	2682#	2692#	2732#	2759#	2966#
	2998#	3056#	3159#	3321#	3508#	3635#	3695#	3715#	3730#	3743#	3807#	3849#	3912#
	4017#	4100#	4159#	4236#	4315#	4337#	4338	4387#	4390#	4440#	4441	4499#	4500
	4516#	4517	4532#	4533	4538#	4565#	4566	4580#	4581	4615#	4867#	4868	4894#
	4895	4913#	4914	4942#	4943	4974#	4975	4987#	4988	5105#	5224#	5225	5261#
	5262	5282#	5283	5314#	5315	5335#	5336	5339#	5363#	5364	5382#	5383	5386#
	5393#	5418#	5419	5437#	5438	5441#	5448#	5474#	5475	5484#	5485	5503#	5504
	5507#	5514#	5590#	5591	5603#	5604	5760#	5768#	5832#	5840#	5904#	5912#	5916#
	5941#	5942	5960#	5961	5964#	5971#	5996#	5997	6015#	6016	6019#	6026#	6051#
	6052	6070#	6071	6074#	6080#	6111#	6112	6131#	6132	6135#	6142#	6168#	6169
	6183#	6184	6194#	6195	6202#	6203	6214#	6215	6239#	6240	6250#	6251	6264#
	6291#	6292	6305#	6306	6313#	6314	6332#	6356#	6357	6370#	6371	6378#	6379
	6397#	6421#	6422	6438#	6439	6446#	6447	6464#	6488#	6489	6502#	6503	6510#
	6511	6529#	6553#	6554	6567#	6568	6575#	6576	6594#	6618#	6619	6634#	6635
	6642#	6643	6661#	6727#	6728	6736#	6737	6741#	6742	6751#	6752	6760#	6761
	6769#	6770	6778#	6779	6787#	6788	6812#	6813	6817#	6818	6827#	6828	6836#
	6837	6845#	6846	6854#	6855	6866#	6867	6876#	6877	6886#	6887	6897#	6898
	6913#	6914	6925#	6926	6936#	6937	6945#	6946	6961#	6962	6973#	6974	6982#
	6983	6991#	6992	7000#	7001	7012#	7013	7022#	7023	7032#	7033	7042#	7043
	7051#	7052	7060#	7061	7065#	7066	7075#	7076	7084#	7085	7099#	7100	7104#
	7105	7114#	7115	7123#	7124	7140#	7141	7151#	7152	7164#	7165	7174#	7175
	7191#	7192	7206#	7207	7225#	7226	7230#	7231	7240#	7241	7249#	7250	7267#
	7268	7272#	7273	7282#	7283	7291#	7292	7300#	7301	7313#	7314	7318#	7319
	7328#	7329	7337#	7338	7346#	7347	7359#	7360	7364#	7365	7374#	7375	7383#
	7384	7392#	7393	7401#	7402	7411#	7425#	7426	7434#	7435	7442#	7443	7453#
	7454	7462#	7463	7471#	7472	7480#	7481	7489#	7490	7508#	7509	7513#	7514
	7524#	7525	7533#	7534	7542#	7543	7556#	7557	7565#	7566	7574#	7575	7585#
	7586	7595#	7596	7604#	7605	7613#	7614	7622#	7623	7631#	7632	7644#	7645
	7655#	7656	7670#	7671	7680#	7681	7692#	7693	7701#	7702	7710#	7711	7722#
	7723	7732#	7733	7742#	7743	7751#	7752	7760#	7761	7769#	7770	7778#	7779



CVDMAA.P11

12-DEC-80 15:59

CROSS REFERENCE TABLE -- USER SYMBOLS

7783#	7784	7793#	7794	7802#	7803	7811#	7812	7820#	7821	7836#	7837	7841#
7842	7852#	7853	7861#	7862	7870#	7871	7887#	7888	7899#	7900	7912#	7913
7923#	7924	7940#	7941	7955#	7956	7974#	7975	7979#	7980	7999#	8000	8008#
8009	8013#	8014	8023#	8024	8032#	8033	8041#	8042	8049#	8050	8066#	8067
8071#	8072	8081#	8082	8090#	8091	8103#	8104	8108#	8109	8118#	8119	8127#
8128	8143#	8144	8148#	8149	8158#	8159	8167#	8168	8176#	8177	8185#	8186
8194#	8195	8203#	8206#	8375#	8376	8384#	8385	8389#	8390	8399#	8400	8413#
8414	8424#	8425	8434#	8435	8439#	8440	8453#	8454	8461#	8462	8473#	8474
8478#	8479	8491#	8492	8502#	8503	8512#	8513	8517#	8518	8527#	8528	8537#
8538	8542#	8543	8553#	8564#	8565	8574#	8575	8584#	8585	8593#	8594	8598#
8599	8608#	8609	8622#	8623	8633#	8634	8642#	8643	8652#	8653	8664#	8665
8669#	8670	8679#	8680	8688#	8689	8699#	8700	8708#	8709	8725#	8726	8733#
8734	8745#	8746	8750#	8751	8760#	8761	8766#	8767	8777#	8778	8787#	8788
8792#	8793	8802#	8803	8807#	8808	8817#	8818	8826#	8827	8835#	8836	8844#
8845	8853#	8854	8874#	8875	8879#	8880	8889#	8890	8899#	8902#	8924#	8929#
8934#	8941#	8963#	8976#									
1242#	3767#	3875#	3930#	4037#	4114#	4174#	4264#	4266	4319	4411#	4559#	4861#
5139#	5357#	5371	5412#	5426	5468#	5492	5581#	5613	5773	5845	5935#	5949
5990#	6004	6045#	6059	6105#	6120	6161#	6284#	6349#	6414#	6481#	6546#	6611#
6716#	6718	7418	8366#	8368	8557	8981						
1242#	2587	2621	2652	2664	2677	2683	2693	2709	2726	2733	2745	2755
2760	2774	2793	2803	2823	2833	2853	2863	2872	2891	2901	2921	2931
2951	2961	2967	2981	2994	2999	3019	3027	3035	3043	3051	3057	3082
3090	3113	3121	3129	3138	3146	3154	3160	3195	3216	3224	3253	3261
3278	3288	3305	3315	3322	3355	3369	3376	3390	3397	3411	3418	3432
3447	3457	3464	3474	3486	3522	3528	3534	3540	3557	3562	3580	3620
3628	3636	3668	3687	3692	3696	3710	3713	3716	3728	3731	3744	3776
3795	3802	3808	3826	3832	3838	3846	3850	3882	3907	3913	4012	4018
4044	4085	4101	4123	4139	4153	4160	4186	4194	4217	4230	4237	4267
4271	4310	4316	4320	4332	4337	4382	4388	4391	4416	4426	4435	4440
4494	4499	4511	4516	4527	4532	4539	4563	4565	4578	4580	4586	4596
4601	4613	4616	4865	4867	4892	4894	4911	4913	4933	4940	4942	4956
4972	4974	4985	4987	5001	5103	5106	5143	5151	5158	5176	5191	5219
5224	5234	5241	5248	5256	5261	5270	5277	5282	5290	5297	5309	5314
5322	5330	5335	5340	5361	5363	5372	5380	5382	5387	5394	5416	5418
5427	5435	5437	5442	5449	5472	5474	5482	5484	5493	5501	5503	5508
5515	5588	5590	5601	5603	5614	5623	5642	5663	5686	5712	5729	5752
5761	5769	5774	5790	5808	5824	5833	5841	5846	5862	5880	5896	5905
5913	5917	5939	5941	5950	5958	5960	5965	5972	5994	5996	6005	6013
6015	6020	6027	6049	6051	6060	6068	6070	6075	6081	6109	6111	6121
6129	6131	6136	6143	6166	6168	6181	6183	6192	6194	6200	6202	6212
6214	6222	6237	6239	6248	6250	6258	6265	6289	6291	6303	6305	6311
6313	6326	6333	6354	6356	6368	6370	6376	6378	6391	6398	6419	6421
6436	6438	6444	6446	6458	6465	6486	6488	6500	6502	6508	6510	6523
6530	6551	6553	6565	6567	6573	6575	6588	6595	6616	6618	6632	6634
6640	6642	6655	6662	6719	6725	6727	6734	6736	6741	6749	6751	6758
6760	6767	6769	6776	6778	6785	6787	6791	6799	6810	6812	6817	6825
6827	6834	6836	6843	6845	6849	6854	6864	6866	6874	6876	6884	6886
6897	6902	6911	6913	6923	6925	6934	6936	6943	6945	6949	6959	6961
6971	6973	6980	6982	6989	6991	6995	7000	7010	7012	7020	7022	7032
7040	7042	7049	7051	7055	7060	7065	7073	7075	7082	7084	7088	7097
7099	7104	7112	7114	7121	7123	7127	7138	7140	7149	7151	7155	7164
7172	7174	7178	7189	7191	7197	7206	7211	7223	7225	7230	7238	7240
7247	7249	7253	7265	7267	7272	7280	7282	7289	7291	7295	7300	7311
7313	7318	7326	7328	7335	7337	7341	7346	7357	7359	7364	7372	7374
7381	7383	7390	7392	7399	7401	7405	7412	7419	7423	7425	7432	7434

T\$TEST= 000034

T\$TSTM= 177777



CVDMAA.P11 12-DEC-80 15:59

CROSS REFERENCE TABLE -- USER SYMBOLS

7442	7451	7453	7460	7462	7469	7471	7478	7480	7487	7489	7493	7506	
7508	7513	7522	7524	7531	7533	7537	7542	7554	7556	7563	7565	7572	
7574	7585	7593	7595	7602	7604	7611	7613	7620	7622	7629	7631	7635	
7642	7644	7653	7655	7661	7668	7670	7678	7680	7690	7692	7699	7701	
7705	7710	7720	7722	7732	7740	7742	7749	7751	7758	7760	7767	7769	
7773	7778	7783	7791	7793	7800	7802	7809	7811	7818	7820	7824	7834	
7836	7841	7850	7852	7859	7861	7868	7870	7874	7885	7887	7897	7899	
7903	7912	7921	7923	7927	7938	7940	7946	7955	7960	7972	7974	7979	
7984	7997	7999	8006	8008	8013	8021	8023	8030	8032	8036	8041	8049	
8054	8064	8066	8071	8079	8081	8085	8090	8101	8103	8108	8116	8118	
8125	8127	8131	8141	8143	8148	8156	8158	8165	8167	8174	8176	8183	
8185	8192	8194	8198	8204	8207	8242	8306	8369	8373	8375	8382	8384	
8389	8394	8399	8413	8419	8424	8432	8434	8439	8444	8451	8453	8461	
8471	8473	8478	8483	8491	8497	8502	8510	8512	8517	8522	8527	8535	
8537	8542	8547	8554	8558	8562	8564	8572	8574	8582	8584	8591	8593	
8598	8603	8608	8622	8628	8633	8642	8647	8652	8662	8664	8669	8674	
8679	8688	8694	8699	8708	8713	8723	8725	8733	8743	8745	8750	8755	
8760	8766	8772	8777	8785	8787	8792	8797	8802	8807	8815	8817	8824	
8826	8833	8835	8842	8844	8851	8853	8857	8872	8874	8879	8884	8889	
8896	8900	8903											
T\$TSTS= 000001	1242#	3768#	3876#	3931#	4038#	4115#	4175#	4265#	4412#	4560#	4862#	5140#	5358#
	5413#	5469#	5582#	5936#	5991#	6046#	6106#	6162#	6285#	6350#	6415#	6482#	6547#
	6612#	6717#	8367#										
T\$SAU = 010023	3741#	3743											
T\$SAUT= 010020	3662#	3695											
T\$SCLE= 010021	3707#	3715											
T\$SDU = 010022	3725#	3730											
T\$SHAR= 010101	8920#	8942											
T\$SHW = 010000	1406#	1428											
T\$SINI= 010017	3516#	3635											
T\$MSG= 010025	2648#	2651	2657#	2682	2688#	2692	2698#	2732	2738#	2759	2765#	2966	2972#
	2998	3004#	3056	3106#	3159	3236#	3321	3820#	3849				
T\$SPRO= 010016	3503#												
T\$SSEG= 010000	5623#	5760#	5790#	5832#	5862#	5904#							
T\$SOF= 010102	8960#	8964											
T\$SRV= 010003	2580#	2599	2614#	2633									
T\$SUB= 010100	4267#	4315	4320#	4387	5372#	5386	5427#	5441	5493#	5507	5614#	5768	5774#
	5840	5846#	5912	5950#	5964	6005#	6019	6060#	6074	6121#	6135	6719#	6741
	6817	6854	6897	7000	7032	7060	7065	7104	7164	7206	7230	7272	7300
	7318	7346	7364	7411	7419#	7442	7513	7542	7585	7710	7732	7778	7783
	7841	7912	7955	7979	8013	8041	8049	8071	8090	8108	8148	8203	8369#
	8389	8399	8413	8424	8439	8461	8478	8491	8502	8517	8527	8542	8553
	8558#	8598	8608	8622	8633	8642	8652	8669	8679	8688	8699	8708	8733
	8750	8760	8766	8777	8792	8802	8807	8879	8889	8899			
T\$SSW = 010001	1437#	1441											
T\$STES= 010076	3768#	3807	3876#	3912	3931#	4017	4038#	4100	4115#	4159	4175#	4236	4265#
	4337	4390	4412#	4440	4499	4516	4532	4538	4560#	4565	4580	4615	4862#
	4867	4894	4913	4942	4974	4987	5105	5140#	5224	5261	5282	5314	5335
	5339	5358#	5363	5382	5393	5413#	5418	5437	5448	5469#	5474	5484	5503
	5514	5582#	5590	5603	5916	5936#	5941	5960	5971	5991#	5996	6015	6026
	6046#	6051	6070	6080	6106#	6111	6131	6142	6162#	6168	6183	6194	6202
	6214	6239	6250	6264	6285#	6291	6305	6313	6332	6350#	6356	6370	6378
	6397	6415#	6421	6438	6446	6464	6482#	6488	6502	6510	6529	6547#	6553
	6567	6575	6594	6612#	6618	6634	6642	6661	6717#	6727	6736	6751	6760
	6769	6778	6787	6812	6827	6836	6845	6866	6876	6886	6913	6925	6936
	6945	6961	6973	6982	6991	7012	7022	7042	7051	7075	7084	7099	7114







CVDMAA.P11 12-DEC-80 15:59

CROSS REFERENCE TABLE -- USER SYMBOLS

WSR14	002262	1703#	2431*	3468										
WSR16	002264	1705#	2432*	3467										
WSR2	002250	1693#	2426*	3452	4353	4358								
WSR4	002252	1695#	2427*	3451	4363	4367								
WSR6	002254	1697#	2428*	3450	4371	4374								
W0	= 003322	2078#	2079											
W1	= 003324	2079#	2080											
W2	= 003326	2080#	2081											
W3	= 003330	2081#	2082											
W4	= 003332	2082#	2083											
W5	= 003334	2083#	2084											
W6	= 003336	2084#	2085											
W7	= 003340	2085#	2086											
W8	= 003342	2086#	2087											
W9	= 003344	2087#	2088											
XDATA	002314	1722#	2669	2749	2986	3338*	3339*							
XORGB	011276	2666	2747	2983	3336#									
X\$ALWA=	000000	1242#												
X\$FALS=	000040	1242#												
X\$OFFS=	000400	1242#												
X\$TRUE=	000020	1242#												
SE	= 000146	1670#	2191#	2231#	2271#	2314#	2377#	2562#	2586#	2620#	3801#	3906#	4011#	4084#
		4152#	4216#	4229#	4309#	4331#	4381#	4434#	4493#	4510#	4526#	4660#	4702#	4955#
		5000#	5218#	5255#	5276#	5308#	5329#	5641#	5662#	5685#	5711#	5728#	5751#	5807#
		5823#	5879#	5895#	6221#	6257#	6325#	6390#	6457#	6522#	6587#	6654#	6790#	6848#
		6901#	6948#	6994#	7054#	7087#	7126#	7154#	7177#	7196#	7210#	7252#	7294#	7340#
		7404#	7492#	7536#	7634#	7660#	7704#	7772#	7823#	7873#	7902#	7926#	7945#	7959#
		7983#	8035#	8053#	8084#	8130#	8197#	8280#	8393#	8418#	8443#	8482#	8496#	8521#
		8546#	8602#	8627#	8646#	8673#	8693#	8712#	8754#	8771#	8796#	8856#	8883#	
		1246#												
SLSTIN=	000001	1247#												
SLSTTA=	000001	1779#	1784	1785	1787	1788	1790	1791	1793	1794	1796	1797	1799	1800
\$MPCSR=	160000 G	1802	1803	1805	1806									
ST	= 000034	1670#	3745#	3854#	3914#	4019#	4102#	4161#	4238#	4392#	4540#	4785#	5109#	5341#
		5395#	5450#	5516#	5918#	5973#	6028#	6082#	6144#	6266#	6334#	6399#	6466#	6531#
		6596#	6664#	8316#										
.	= 040344	1238#	1896#	2068#	2072#	2113#	2129#	2203#	3095#	3853#	4338	4441	4500	4517
		4533	4566	4581	4611	4868	4890	4895	4909	4914	4938	4943	4970	4975
		4983	4988	5101	5225	5262	5283	5315	5336	5364	5383	5419	5438	5475
		5485	5504	5591	5599	5604	5942	5961	5997	6016	6052	6071	6112	6132
		6169	6184	6195	6203	6215	6240	6251	6292	6306	6314	6357	6371	6379
		6422	6439	6447	6489	6503	6511	6554	6568	6576	6619	6635	6643	6728
		6732	6737	6739	6742	6747	6752	6756	6761	6765	6770	6774	6779	6783
		6788	6808	6813	6815	6818	6823	6828	6832	6837	6841	6846	6855	6862
		6867	6872	6877	6882	6887	6895	6898	6909	6914	6921	6926	6932	6937
		6941	6946	6957	6962	6969	6974	6978	6983	6987	6992	7001	7008	7013
		7018	7023	7030	7033	7038	7043	7047	7052	7061	7063	7066	7071	7076
		7080	7085	7095	7100	7102	7105	7110	7115	7119	7124	7136	7141	7147
		7152	7162	7165	7170	7175	7187	7192	7204	7207	7221	7226	7228	7231
		7236	7241	7245	7250	7263	7268	7270	7273	7278	7283	7287	7292	7301
		7309	7314	7316	7319	7324	7329	7333	7338	7347	7355	7360	7362	7365
		7370	7375	7379	7384	7388	7393	7397	7402	7426	7430	7435	7440	7443
		7449	7454	7458	7463	7467	7472	7476	7481	7485	7490	7504	7509	7511
		7514	7520	7525	7529	7534	7543	7552	7557	7561	7566	7570	7575	7583
		7586	7591	7596	7600	7605	7609	7614	7618	7623	7627	7632	7640	7645
		7651	7656	7671	7676	7681	7688	7693	7697	7702	7711	7718	7723	7730



CVDMAA.P11

12-DEC-80 15:59

CROSS REFERENCE TABLE -- USER SYMBOLS

7733	7738	7743	7747	7752	7756	7761	7765	7770	7779	7781	7784	7789
7794	7798	7803	7807	7812	7816	7821	7832	7837	7842	7848	7853	7857
7862	7866	7871	7883	7888	7895	7900	7910	7913	7919	7924	7936	7941
7953	7956	7970	7975	7980	7995	8000	8004	8009	8011	8014	8019	8024
8028	8033	8042	8047	8050	8062	8067	8069	8072	8077	8082	8091	8099
8104	8106	8109	8114	8119	8123	8128	8139	8144	8146	8149	8154	8159
8163	8168	8172	8177	8181	8186	8190	8195	8376	8380	8385	8387	8390
8400	8411	8414	8425	8430	8435	8437	8440	8449	8454	8459	8462	8469
8474	8476	8479	8489	8492	8503	8508	8513	8515	8518	8528	8533	8538
8540	8543	8565	8570	8575	8580	8585	8589	8594	8596	8599	8609	8620
8623	8634	8640	8643	8653	8660	8665	8667	8670	8680	8686	8689	8700
8706	8709	8721	8726	8731	8734	8741	8746	8748	8751	8761	8764	8767
8778	8783	8788	8790	8793	8803	8805	8808	8813	8818	8822	8827	8831
8836	8840	8845	8849	8854	8870	8875	8877	8880	8890	8894	8967#	





CVDMAA.P11 12-DEC-80 15:59

CROSS REFERENCE TABLE -- MACRO NAMES

ENDSRV	1#	1242#	2598	2632											
ENDSUB	1#	1242#	4314	4386	5385	5440	5506	5767	5839	5911	5963	6018	6073	6134	7410
	8202	8552	8898												
ENDSW	1#	1242#	1440												
ENDTST	1#	1242#	3806	3911	4016	4099	4158	4235	4389	4537	4614	5104	5338	5392	5447
	5513	5915	5970	6025	6079	6141	6263	6331	6396	6463	6528	6593	6660	8205	8901
EQUALS	1#	1242#	1450												
ERRDF	1#	1242#	2587	2621	3802	3907	4012	4085	4153	4217	4230	4310	4332	4382	4435
	4494	4511	4527	4956	5001	5219	5256	5277	5309	5330	5642	5663	5686	5712	5729
	5752	5808	5824	5880	5896	6222	6258	6326	6391	6458	6523	6588	6655	6791	6849
	6902	6949	6995	7055	7088	7127	7155	7178	7197	7211	7253	7295	7341	7405	7493
	7537	7635	7661	7705	7773	7824	7874	7903	7927	7946	7960	7984	8036	8054	8085
	8131	8198	8394	8419	8444	8483	8497	8522	8547	8603	8628	8647	8674	8694	8713
	8755	8772	8797	8857	8884										
ERRHRD	1#	1242#													
ERROR	1#	1242#	3881	4043	4122	4138	4185	4270	4415	4425	4562	4577	4595	4612	4864
	4891	4910	4939	4971	4984	5102	5142	5150	5157	5175	5190	5233	5240	5247	5269
	5289	5296	5321	5360	5379	5415	5434	5471	5481	5500	5587	5600	5938	5957	5993
	6012	6048	6067	6108	6128	6165	6180	6191	6199	6211	6236	6247	6288	6302	6310
	6353	6367	6375	6418	6435	6443	6485	6499	6507	6550	6564	6572	6615	6631	6639
	6724	6733	6748	6757	6766	6775	6784	6809	6824	6833	6842	6863	6873	6883	6910
	6922	6933	6942	6958	6970	6979	6988	7009	7019	7039	7048	7072	7081	7096	7111
	7120	7137	7148	7171	7188	7222	7237	7246	7264	7279	7288	7310	7325	7334	7356
	7371	7380	7389	7398	7422	7431	7450	7459	7468	7477	7486	7505	7521	7530	7553
	7562	7571	7592	7601	7610	7619	7628	7641	7652	7667	7677	7689	7698	7719	7739
	7748	7757	7766	7790	7799	7808	7817	7833	7849	7858	7867	7884	7896	7920	7937
	7971	7996	8005	8020	8029	8063	8078	8100	8115	8124	8140	8155	8164	8173	8182
	8191	8241	8305	8372	8381	8431	8450	8470	8509	8534	8561	8571	8581	8590	8661
	8722	8742	8784	8814	8823	8832	8841	8850	8871	8895					
ERRSF	1#	1242#													
ERRSOF	1#	1242#													
ERRTBL	1#	1242#	1681												
ESCAPE	1#	1242#	4336	4439	4498	4515	4531	4564	4579	4866	4893	4912	4941	4973	4986
	5223	5260	5281	5313	5334	5362	5381	5417	5436	5473	5483	5502	5589	5602	5940
	5959	5995	6014	6050	6069	6110	6130	6167	6182	6193	6201	6213	6238	6249	6290
	6304	6312	6355	6369	6377	6420	6437	6445	6487	6501	6509	6552	6566	6574	6617
	6633	6641	6726	6735	6740	6750	6759	6768	6777	6786	6811	6816	6826	6835	6844
	6853	6865	6875	6885	6896	6912	6924	6935	6944	6960	6972	6981	6990	6999	7011
	7021	7031	7041	7050	7059	7064	7074	7083	7098	7103	7113	7122	7139	7150	7163
	7173	7190	7205	7224	7229	7239	7248	7266	7271	7281	7290	7299	7312	7317	7327
	7336	7345	7358	7363	7373	7382	7391	7400	7424	7433	7441	7452	7461	7470	7479
	7488	7507	7512	7523	7532	7541	7555	7564	7573	7584	7594	7603	7612	7621	7630
	7643	7654	7669	7679	7691	7700	7709	7721	7731	7741	7750	7759	7768	7777	7782
	7792	7801	7810	7819	7835	7840	7851	7860	7869	7886	7898	7911	7922	7939	7954
	7973	7978	7998	8007	8012	8022	8031	8040	8048	8065	8070	8080	8089	8102	8107
	8117	8126	8142	8147	8157	8166	8175	8184	8193	8374	8383	8388	8398	8412	8423
	8433	8438	8452	8460	8472	8477	8490	8501	8511	8516	8526	8536	8541	8563	8573
	8583	8592	8597	8607	8621	8632	8641	8651	8663	8668	8678	8687	8698	8707	8724
	8732	8744	8749	8759	8765	8776	8786	8791	8801	8806	8816	8825	8834	8843	8852
	8873	8878	8888												
EXIT	1#	1242#													
FEQUAL	1#	1242#													
GEDF	1670#	2585	2619	3800	3905	4010	4083	4151	4215	4228	4308	4330	4380	4433	4492
	4509	4525	4954	4999	5217	5254	5275	5307	5328	5640	5661	5684	5710	5727	5750
	5806	5822	5878	5894	6220	6256	6324	6389	6456	6521	6586	6653	6789	6847	6900
	6947	6993	7053	7086	7125	7153	7176	7195	7209	7251	7293	7339	7403	7491	7535

CVDMAA.P11 12-DEC-80 15:59

CROSS REFERENCE TABLE -- MACRO NAMES

	7633	7659	7703	7771	7822	7872	7901	7925	7944	7958	7982	8034	8052	8083	8129
	8196	8392	8417	8442	8481	8495	8520	8545	8601	8626	8645	8672	8692	8711	8753
	8770	8795	8855	8882											
GEHRD	1670#														
GESF	1670#														
GESFT	1670#														
GETBYT	1#	1242#													
GETPRI	1#	1242#													
GETWOR	1#	1242#													
GPIAIA	1#	1242#													
GPIAID	1#	1242#													
GPIAIL	1#	1242#													
GPHARD	1#	1242#	3578												
GPRMA	1#	1242#	8923	8928											
GPRMD	1#	1242#	8933												
GPRML	1#	1242#													
GTDF	1670#	2190	2230	2270	2313	2376	2561	4659	4701	8279					
GTHRD	1670#														
GTSF	1670#														
GTSFT	1670#														
HEADER	1#	1242#	1270												
INLOOP	1#	1242#													
IOSETU	1#	1242#													
IOSTAR	1#	1242#													
KT11	1#	1242#													
LASTAD	1#	1242#	8976												
MANUAL	1#	1242#													
MEMORY	1#	1242#													
MSG	3745#	3751	3854#	3860	3914#	3920	4019#	4025	4102#	4108	4161#	4167	4238#	4244	4392#
	4398	4540#	4546	4785#	4791	5109#	5115	5341#	5347	5395#	5401	5450#	5456	5516#	5522
	5918#	5924	5973#	5979	6028#	6034	6082#	6088	6144#	6150	6266#	6272	6334#	6340	6399#
	6405	6466#	6472	6531#	6537	6596#	6602	6664#	6670	8316#	8322				
MSBYTE	1#	1242#	1271#	1277	1278	1279									
MSCHEC	1#	1242#													
MSCNTO	1#	1242#	8924#	8929#	8934#										
MSCOUN	1#	1242#	2659#	2668#	2704#	2720#	2740#	2749#	2770#	2777#	2796#	2807#	2826#	2837#	2856#
	2868#	2875#	2894#	2905#	2924#	2935#	2954#	2976#	2985#	3015#	3024#	3032#	3040#	3048#	3072#
	3087#	3110#	3118#	3126#	3135#	3143#	3151#	3184#	3208#	3221#	3249#	3258#	3266#	3281#	3293#
	3308#	3350#	3358#	3372#	3379#	3393#	3400#	3414#	3421#	3442#	3450#	3460#	3467#	3483#	3822#
	3829#	3835#	3841#	6796#											
MSDATA	1#	1242#	1271#	1280	1282	1284	1286	1288	1290	1292	1294	1296	1298	1300	1302
	1304	1306	1308	1310#	1312	1314	1317	1320	1322	1324	1326	1328	1330	1332	1334
	1336	1338	1340	1342	1344	1346	1348	1350	1352	1354	2109#	2121#			
MSDECR	1#	1242#	1428#	1441#	2599#	2633#	2651#	2682#	2692#	2732#	2759#	2966#	2998#	3056#	3159#
	3321#	3508#	3635#	3695#	3715#	3730#	3743#	3807#	3849#	3912#	4017#	4100#	4159#	4236#	4315#
	4387#	4390#	4538#	4615#	5105#	5339#	5386#	5393#	5441#	5448#	5507#	5514#	5760#	5768#	5832#
	5840#	5904#	5912#	5916#	5964#	5971#	6019#	6026#	6074#	6080#	6135#	6142#	6264#	6332#	6397#
	6464#	6529#	6594#	6661#	7411#	8203#	8206#	8553#	8899#	8902#	8941#	8963#	8976#		
MSDEFA	1#	1242#	8924#	8929#	8934#										
MSENDE	1#	1242#	1428#	1441#	2599#	2633#	2651#	2682#	2692#	2732#	2759#	2966#	2998#	3056#	3159#
	3321#	3635#	3695#	3715#	3730#	3743#	3807#	3849#	3912#	4017#	4100#	4159#	4236#	4315#	4387#
	4390#	4538#	4615#	5105#	5339#	5386#	5393#	5441#	5448#	5507#	5514#	5760#	5768#	5832#	5840#
	5904#	5912#	5916#	5964#	5971#	6019#	6026#	6074#	6080#	6135#	6142#	6264#	6332#	6397#	6464#
	6529#	6594#	6661#	7411#	8203#	8206#	8553#	8899#	8902#	8941#	8963#	8976#			
MSERRI	1#	1242#	2587#	2621#	3802#	3907#	4012#	4085#	4153#	4217#	4230#	4310#	4332#	4382#	4435#
	4494#	4511#	4527#	4956#	5001#	5219#	5256#	5277#	5309#	5330#	5642#	5663#	5686#	5712#	5729#



CVDMAA.P11 12-DEC-80 15:59

CROSS REFERENCE TABLE -- MACRO NAMES

	5752#	5808#	5824#	5880#	5896#	6222#	6258#	6326#	6391#	6458#	6523#	6588#	6655#	6791#	6849#
	6902#	6949#	6995#	7055#	7088#	7127#	7155#	7178#	7197#	7211#	7253#	7295#	7341#	7405#	7493#
	7537#	7635#	7661#	7705#	7773#	7824#	7874#	7903#	7927#	7946#	7960#	7984#	8036#	8054#	8085#
	8131#	8198#	8394#	8419#	8444#	8483#	8497#	8522#	8547#	8603#	8628#	8647#	8674#	8694#	8713#
MSESCA	1#	1242#	4337#	4338	4440#	4441	4499#	4500	4516#	4517	4532#	4533	4565#	4566	4580#
	4581	4867#	4868	4894#	4895	4913#	4914	4942#	4943	4974#	4975	4987#	4988	5224#	5225
	5261#	5262	5282#	5283	5314#	5315	5335#	5336	5363#	5364	5382#	5383	5418#	5419	5437#
	5438	5474#	5475	5484#	5485	5503#	5504	5590#	5591	5603#	5604	5941#	5942	5960#	5961
	5996#	5997	6015#	6016	6051#	6052	6070#	6071	6111#	6112	6131#	6132	6168#	6169	6183#
	6184	6194#	6195	6202#	6203	6214#	6215	6239#	6240	6250#	6251	6291#	6292	6305#	6306
	6313#	6314	6356#	6357	6370#	6371	6378#	6379	6421#	6422	6438#	6439	6446#	6447	6488#
	6489	6502#	6503	6510#	6511	6553#	6554	6567#	6568	6575#	6576	6618#	6619	6634#	6635
	6642#	6643	6727#	6728	6736#	6737	6741#	6742	6751#	6752	6760#	6761	6769#	6770	6778#
	6779	6787#	6788	6812#	6813	6817#	6818	6827#	6828	6836#	6837	6845#	6846	6854#	6855
	6866#	6867	6876#	6877	6886#	6887	6897#	6898	6913#	6914	6925#	6926	6936#	6937	6945#
	6946	6961#	6962	6973#	6974	6982#	6983	6991#	6992	7000#	7001	7012#	7013	7022#	7023
	7032#	7033	7042#	7043	7051#	7052	7060#	7061	7065#	7066	7075#	7076	7084#	7085	7099#
	7100	7104#	7105	7114#	7115	7123#	7124	7140#	7141	7151#	7152	7164#	7165	7174#	7175
	7191#	7192	7206#	7207	7225#	7226	7230#	7231	7240#	7241	7249#	7250	7267#	7268	7272#
	7273	7282#	7283	7291#	7292	7300#	7301	7313#	7314	7318#	7319	7328#	7329	7337#	7338
	7346#	7347	7359#	7360	7364#	7365	7374#	7375	7383#	7384	7392#	7393	7401#	7402	7425#
	7426	7434#	7435	7442#	7443	7453#	7454	7462#	7463	7471#	7472	7480#	7481	7489#	7490
	7508#	7509	7513#	7514	7524#	7525	7533#	7534	7542#	7543	7556#	7557	7565#	7566	7574#
	7575	7585#	7586	7595#	7596	7604#	7605	7613#	7614	7622#	7623	7631#	7632	7644#	7645
	7655#	7656	7670#	7671	7680#	7681	7692#	7693	7701#	7702	7710#	7711	7722#	7723	7732#
	7733	7742#	7743	7751#	7752	7760#	7761	7769#	7770	7778#	7779	7783#	7784	7793#	7794
	7802#	7803	7811#	7812	7820#	7821	7836#	7837	7841#	7842	7852#	7853	7861#	7862	7870#
	7871	7887#	7888	7899#	7900	7912#	7913	7923#	7924	7940#	7941	7955#	7956	7974#	7975
	7979#	7980	7999#	8000	8008#	8009	8013#	8014	8023#	8024	8032#	8033	8041#	8042	8049#
	8050	8066#	8067	8071#	8072	8081#	8082	8090#	8091	8103#	8104	8108#	8109	8118#	8119
	8127#	8128	8143#	8144	8148#	8149	8158#	8159	8167#	8168	8176#	8177	8185#	8186	8194#
	8195	8375#	8376	8384#	8385	8389#	8390	8399#	8400	8413#	8414	8424#	8425	8434#	8435
	8439#	8440	8453#	8454	8461#	8462	8473#	8474	8478#	8479	8491#	8492	8502#	8503	8512#
	8513	8517#	8518	8527#	8528	8537#	8538	8542#	8543	8564#	8565	8574#	8575	8584#	8585
	8593#	8594	8598#	8599	8608#	8609	8622#	8623	8633#	8634	8642#	8643	8652#	8653	8664#
	8665	8669#	8670	8679#	8680	8688#	8689	8699#	8700	8708#	8709	8725#	8726	8733#	8734
	8745#	8746	8750#	8751	8760#	8761	8766#	8767	8777#	8778	8787#	8788	8792#	8793	8802#
	8803	8807#	8808	8817#	8818	8826#	8827	8835#	8836	8844#	8845	8853#	8854	8874#	8875
	8879#	8880	8889#	8890											
MSESCS	1#	1242#	4337#	4440#	4499#	4516#	4532#	4565#	4580#	4867#	4894#	4913#	4942#	4974#	4987#
	5224#	5261#	5282#	5314#	5335#	5363#	5382#	5418#	5437#	5474#	5484#	5503#	5590#	5603#	5941#
	5960#	5996#	6015#	6051#	6070#	6111#	6131#	6168#	6183#	6194#	6202#	6214#	6239#	6250#	6291#
	6305#	6313#	6356#	6370#	6378#	6421#	6438#	6446#	6488#	6502#	6510#	6553#	6567#	6575#	6618#
	6634#	6642#	6727#	6736#	6741#	6751#	6760#	6769#	6778#	6787#	6812#	6817#	6827#	6836#	6845#
	6854#	6866#	6876#	6886#	6897#	6913#	6925#	6936#	6945#	6961#	6973#	6982#	6991#	7000#	7012#
	7022#	7032#	7042#	7051#	7060#	7065#	7075#	7084#	7099#	7104#	7114#	7123#	7140#	7151#	7164#
	7174#	7191#	7206#	7225#	7230#	7240#	7249#	7267#	7272#	7282#	7291#	7300#	7313#	7318#	7328#
	7337#	7346#	7359#	7364#	7374#	7383#	7392#	7401#	7425#	7434#	7442#	7453#	7462#	7471#	7480#
	7489#	7508#	7513#	7524#	7533#	7542#	7556#	7565#	7574#	7585#	7595#	7604#	7613#	7622#	7631#
	7644#	7655#	7670#	7680#	7692#	7701#	7710#	7722#	7732#	7742#	7751#	7760#	7769#	7778#	7783#
	7793#	7802#	7811#	7820#	7836#	7841#	7852#	7861#	7870#	7887#	7899#	7912#	7923#	7940#	7955#
	7974#	7979#	7999#	8008#	8013#	8023#	8032#	8041#	8049#	8066#	8071#	8081#	8090#	8103#	8108#
	8118#	8127#	8143#	8148#	8158#	8167#	8176#	8185#	8194#	8375#	8384#	8389#	8399#	8413#	8424#
	8434#	8439#	8453#	8461#	8473#	8478#	8491#	8502#	8512#	8517#	8527#	8537#	8542#	8564#	8574#
	8584#	8593#	8598#	8608#	8622#	8633#	8642#	8652#	8664#	8669#	8679#	8688#	8699#	8708#	8725#



CVDMAA.P11 12-DEC-80 15:59

CROSS REFERENCE TABLE -- MACRO NAMES

	8733#	8745#	8750#	8760#	8766#	8777#	8787#	8792#	8802#	8807#	8817#	8826#	8835#	8844#	8853#
	8874#	8879#	8889#												
MSEXCP	1#	1242#	8924#	8929#	8934#										
MSEXIT	1#	1242#													
MSEXSE	1#	1242#													
MSEXTJ	1#	1242#													
MSGEN	1#	1242#	1244#	1271#	1280#	1282#	1284#	1286#	1288#	1290#	1292#	1294#	1296#	1298#	1300#
	1302#	1304#	1306#	1308#	1310#	1312#	1314#	1317#	1320#	1322#	1324#	1326#	1328#	1330#	1332#
	1334#	1336#	1338#	1340#	1342#	1344#	1346#	1348#	1350#	1352#	1354#	1367#	1407#	1408#	1428#
	1438#	1439#	1441#	1682#	2109#	2121#	2580#	2599#	2614#	2633#	2648#	2651#	2657#	2682#	2688#
	2692#	2698#	2732#	2738#	2759#	2765#	2966#	2972#	2998#	3004#	3056#	3106#	3159#	3236#	3321#
	3503#	3516#	3635#	3662#	3695#	3707#	3715#	3725#	3730#	3741#	3743#	3767#	3807#	3820#	3849#
	3875#	3912#	3930#	4017#	4037#	4100#	4114#	4159#	4174#	4236#	4264#	4266#	4315#	4319#	4387#
	4390#	4411#	4538#	4559#	4615#	4861#	5105#	5139#	5339#	5357#	5371#	5386#	5393#	5412#	5426#
	5441#	5448#	5468#	5492#	5507#	5514#	5581#	5613#	5760#	5768#	5773#	5832#	5840#	5845#	5904#
	5912#	5916#	5935#	5949#	5964#	5971#	5990#	6004#	6019#	6026#	6045#	6059#	6074#	6080#	6105#
	6120#	6135#	6142#	6161#	6264#	6284#	6332#	6349#	6397#	6414#	6464#	6481#	6529#	6546#	6594#
	6611#	6661#	6716#	6718#	7411#	7418#	8203#	8206#	8366#	8368#	8553#	8557#	8899#	8902#	8921#
	8942#	8961#	8964#	8980#											
MSGENB	1#	1242#													
MSGETS	1#	1242#	1428#	1441#	2599#	2633#	2651#	2682#	2692#	2732#	2759#	2966#	2998#	3056#	3159#
	3321#	3508#	3635#	3695#	3715#	3730#	3743#	3807#	3849#	3912#	4017#	4100#	4159#	4236#	4315#
	4387#	4390#	4538#	4615#	5105#	5339#	5386#	5393#	5441#	5448#	5507#	5514#	5760#	5768#	5832#
	5840#	5904#	5912#	5916#	5964#	5971#	6019#	6026#	6074#	6080#	6135#	6142#	6264#	6332#	6397#
	6464#	6529#	6594#	6661#	7411#	8203#	8206#	8553#	8899#	8902#	8941#	8963#	8976#		
MSGETT	1#	1242#	4337#	4440#	4499#	4516#	4532#	4565#	4580#	4867#	4894#	4913#	4942#	4974#	4987#
	5224#	5261#	5282#	5314#	5335#	5363#	5382#	5418#	5437#	5474#	5484#	5503#	5590#	5603#	5941#
	5960#	5996#	6015#	6051#	6070#	6111#	6131#	6168#	6183#	6194#	6202#	6214#	6239#	6250#	6291#
	6305#	6313#	6356#	6370#	6378#	6421#	6438#	6446#	6488#	6502#	6510#	6553#	6567#	6575#	6618#
	6634#	6642#	6727#	6736#	6741#	6751#	6760#	6769#	6778#	6787#	6812#	6817#	6827#	6836#	6845#
	6854#	6866#	6876#	6886#	6897#	6913#	6925#	6936#	6945#	6961#	6973#	6982#	6991#	7000#	7012#
	7022#	7032#	7042#	7051#	7060#	7065#	7075#	7084#	7099#	7104#	7114#	7123#	7140#	7151#	7164#
	7174#	7191#	7206#	7225#	7230#	7240#	7249#	7267#	7272#	7282#	7291#	7300#	7313#	7318#	7328#
	7337#	7346#	7359#	7364#	7374#	7383#	7392#	7401#	7425#	7434#	7442#	7453#	7462#	7471#	7480#
	7489#	7508#	7513#	7524#	7533#	7542#	7556#	7565#	7574#	7585#	7595#	7604#	7613#	7622#	7631#
	7644#	7655#	7670#	7680#	7692#	7701#	7710#	7722#	7732#	7742#	7751#	7760#	7769#	7778#	7783#
	7793#	7802#	7811#	7820#	7836#	7841#	7852#	7861#	7870#	7887#	7899#	7912#	7923#	7940#	7955#
	7974#	7979#	7999#	8008#	8013#	8023#	8032#	8041#	8049#	8066#	8071#	8081#	8090#	8103#	8108#
	8118#	8127#	8143#	8148#	8158#	8167#	8176#	8185#	8194#	8375#	8384#	8389#	8399#	8413#	8424#
	8434#	8439#	8453#	8461#	8473#	8478#	8491#	8502#	8512#	8517#	8527#	8537#	8542#	8564#	8574#
	8584#	8593#	8598#	8608#	8622#	8633#	8642#	8652#	8664#	8669#	8679#	8688#	8699#	8708#	8725#
	8733#	8745#	8750#	8760#	8766#	8777#	8787#	8792#	8802#	8807#	8817#	8826#	8835#	8844#	8853#
	8874#	8879#	8889#												
MSGNGB	1#	1242#	1244#	1271#	1280#	1282#	1284#	1286#	1288#	1290#	1292#	1294#	1296#	1298#	1300#
	1302#	1304#	1306#	1308#	1310#	1312#	1314#	1317#	1320#	1322#	1324#	1326#	1328#	1330#	1332#
	1334#	1336#	1338#	1340#	1342#	1344#	1346#	1348#	1350#	1352#	1354#	1366#	1367	1406#	1407
	1408	1437#	1438	1439	1682#	2109#	2121#	2580#	2614#	2648#	2657#	2688#	2698#	2738#	2765#
	2972#	3004#	3106#	3236#	3503#	3516#	3662#	3707#	3725#	3741#	3820#	8920#	8921	8960#	8961
	8977#	8980													
MSGNIN	1#	1242#	1271#	1272	1273	1274	1275	1276	1277#	1278#	1279#	1280#	1281	1282#	1283
	1284#	1285	1286#	1287	1288#	1289	1290#	1291	1292#	1293	1294#	1295	1296#	1297	1298#
	1299	1300#	1301	1302#	1303	1304#	1305	1306#	1307	1308#	1309	1310#	1311	1312#	1313
	1314#	1315	1316	1317#	1318	1319#	1320#	1321	1322#	1323	1324#	1325	1326#	1327	1328#
	1329	1330#	1331	1332#	1333	1334#	1335	1336#	1337	1338#	1339	1340#	1341	1342#	1343
	1344#	1345	1346#	1347	1348#	1349	1350#	1351	1352#	1353	1354#	1355	1366#	1368#	1369#
	1370#	1371#	1372#	1373#	1374#	1375#	1376#	1377#	1378#	1379#	1380#	1381#	1382#	1383#	1384#



CVDMAA.P11 12-DEC-80 15:59

CROSS REFERENCE TABLE -- MACRO NAMES

1385#	1386#	1387#	1388#	1389#	1390#	1391#	1392#	1393#	1394#	1395#	1406#	1437#	2109#	2110
2113	2121#	2122	2129	2192	2193	2194	2195	2232	2233	2234	2235	2272	2273	2274
2275	2315	2316	2317	2318	2378	2379	2380	2381	2563	2564	2565	2566	2587#	2588#
2589#	2590#	2599#	2600	2621#	2622#	2623#	2624#	2633#	2634	2652#	2659#	2660#	2661#	2662#
2663	2664#	2665	2668#	2669	2670#	2671	2672#	2673	2674#	2675#	2676	2677#	2678	2683#
2693#	2704#	2705	2706#	2707#	2708	2709#	2710	2720#	2721#	2722	2723#	2724#	2725	2726#
2727	2733#	2740#	2741#	2742#	2743#	2744	2745#	2746	2749#	2750#	2751#	2752#	2753#	2754
2755#	2756	2760#	2770#	2771#	2772#	2773	2774#	2775	2777#	2778	2779#	2780	2781#	2782
2783#	2784	2785#	2786	2787#	2788	2789#	2790#	2791#	2792	2793#	2794	2796#	2797	2798#
2799	2800#	2801#	2802	2803#	2804	2807#	2808	2809#	2810	2811#	2812	2813#	2814	2815#
2816	2817#	2818	2819#	2820#	2821#	2822	2823#	2824	2826#	2827	2828#	2829	2830#	2831#
2832	2833#	2834	2837#	2838	2839#	2840	2841#	2842	2843#	2844	2845#	2846	2847#	2848
2849#	2850#	2851#	2852	2853#	2854	2856#	2857	2858#	2859	2860#	2861#	2862	2863#	2864
2868#	2869#	2870#	2871	2872#	2873	2875#	2876	2877#	2878	2879#	2880	2881#	2882	2883#
2884	2885#	2886	2887#	2888#	2889#	2890	2891#	2892	2894#	2895	2896#	2897	2898#	2899#
2900	2901#	2902	2905#	2906	2907#	2908	2909#	2910	2911#	2912	2913#	2914	2915#	2916
2917#	2918#	2919#	2920	2921#	2922	2924#	2925	2926#	2927	2928#	2929#	2930	2931#	2932
2935#	2936	2937#	2938	2939#	2940	2941#	2942	2943#	2944	2945#	2946	2947#	2948#	2949#
2950	2951#	2952	2954#	2955	2956#	2957	2958#	2959#	2960	2961#	2962	2967#	2976#	2977#
2978#	2979#	2980	2981#	2982	2985#	2986	2987#	2988	2989#	2990	2991#	2992#	2993	2994#
2995	2999#	3015#	3016#	3017#	3018	3019#	3020	3024#	3025#	3026	3027#	3028	3032#	3033#
3034	3035#	3036	3040#	3041#	3042	3043#	3044	3048#	3049#	3050	3051#	3052	3057#	3072#
3073	3074#	3075	3076#	3077	3078#	3079#	3080#	3081	3082#	3083	3087#	3088#	3089	3090#
3091	3110#	3111#	3112	3113#	3114	3118#	3119#	3120	3121#	3122	3126#	3127#	3128	3129#
3130	3135#	3136#	3137	3138#	3139	3143#	3144#	3145	3146#	3147	3151#	3152#	3153	3154#
3155	3160#	3184#	3185	3186#	3187	3188#	3189	3190#	3191#	3192#	3193#	3194	3195#	3196
3208#	3209	3210#	3211#	3212#	3213#	3214#	3215	3216#	3217	3221#	3222#	3223	3224#	3225
3249#	3250#	3251#	3252	3253#	3254	3258#	3259#	3260	3261#	3262	3266#	3267	3268#	3269
3270#	3271	3272#	3273	3274#	3275#	3276#	3277	3278#	3279	3281#	3282	3283#	3284	3285#
3286#	3287	3288#	3289	3293#	3294	3295#	3296	3297#	3298	3299#	3300	3301#	3302#	3303#
3304	3305#	3306	3308#	3309	3310#	3311	3312#	3313#	3314	3315#	3316	3322#	3350#	3351#
3352#	3353#	3354	3355#	3356	3358#	3359	3360#	3361	3362#	3363	3364#	3365	3366#	3367#
3368	3369#	3370	3372#	3373#	3374#	3375	3376#	3377	3379#	3380	3381#	3382	3383#	3384
3385#	3386	3387#	3388#	3389	3390#	3391	3393#	3394#	3395#	3396	3397#	3398	3400#	3401
3402#	3403	3404#	3405	3406#	3407	3408#	3409#	3410	3411#	3412	3414#	3415#	3416#	3417
3418#	3419	3421#	3422	3423#	3424	3425#	3426	3427#	3428	3429#	3430#	3431	3432#	3433
3442#	3443#	3444#	3445#	3446	3447#	3448	3450#	3451#	3452#	3453#	3454#	3455#	3456	3457#
3458	3460#	3461#	3462#	3463	3464#	3465	3467#	3468#	3469#	3470#	3471#	3472#	3473	3474#
3475	3483#	3484#	3485	3486#	3487	3521#	3522#	3524#	3527#	3528#	3530#	3533#	3534#	3536#
3539#	3540#	3542#	3553#	3554#	3555#	3556#	3557#	3558	3561#	3562#	3579#	3580#	3581#	3583#
3616#	3617#	3618#	3619#	3620#	3621	3624#	3625#	3626#	3627#	3628#	3629	3636#	3664#	3665#
3666#	3667#	3668#	3669	3686#	3687#	3691#	3692#	3696#	3709#	3710#	3712#	3713#	3716#	3728#
3731#	3744#	3772#	3773#	3774#	3775#	3776#	3777	3794#	3795#	3802#	3803#	3804#	3805#	3808#
3822#	3823#	3824#	3825	3826#	3827	3829#	3830#	3831	3832#	3833	3835#	3836#	3837	3838#
3839	3841#	3842#	3843#	3844#	3845	3846#	3847	3850#	3882#	3907#	3908#	3909#	3910#	3913#
4012#	4013#	4014#	4015#	4018#	4044#	4085#	4086#	4087#	4088#	4101#	4123#	4139#	4153#	4154#
4155#	4156#	4160#	4186#	4194#	4217#	4218#	4219#	4220#	4230#	4231#	4232#	4233#	4237#	4267#
4271#	4310#	4311#	4312#	4313#	4316#	4320#	4332#	4333#	4334#	4335#	4337#	4338#	4382#	4383#
4384#	4385#	4388#	4391#	4416#	4426#	4435#	4436#	4437#	4438#	4440#	4441#	4494#	4495#	4496#
4497#	4499#	4500#	4511#	4512#	4513#	4514#	4516#	4517#	4527#	4528#	4529#	4530#	4532#	4533#
4539#	4563#	4565#	4566#	4578#	4580#	4581#	4586#	4596#	4601#	4613#	4616#	4661	4662	4663
4664	4703	4704	4705	4706	4865#	4867#	4868#	4892#	4894#	4895#	4911#	4913#	4914#	4933#
4940#	4942#	4943#	4956#	4957#	4958#	4959#	4972#	4974#	4975#	4985#	4987#	4988#	5001#	5002#
5003#	5004#	5103#	5106#	5143#	5151#	5158#	5176#	5191#	5219#	5220#	5221#	5222#	5224#	5225#
5234#	5241#	5248#	5256#	5257#	5258#	5259#	5261#	5262#	5270#	5277#	5278#	5279#	5280#	5282#
5283#	5290#	5297#	5309#	5310#	5311#	5312#	5314#	5315#	5322#	5330#	5331#	5332#	5333#	5335#



CVDMAA.P11

12-DEC-80 15:59

CROSS REFERENCE TABLE -- MACRO NAMES

5336#	5340#	5361#	5363#	5364#	5372#	5380#	5382#	5383#	5387#	5394#	5416#	5418#	5419#	5427#
5435#	5437#	5438#	5442#	5449#	5472#	5474#	5475#	5482#	5484#	5485#	5493#	5501#	5503#	5504#
5508#	5515#	5588#	5590#	5591#	5601#	5603#	5604#	5614#	5623#	5642#	5643#	5644#	5645#	5663#
5664#	5665#	5666#	5686#	5687#	5688#	5689#	5712#	5713#	5714#	5715#	5729#	5730#	5731#	5732#
5752#	5753#	5754#	5755#	5761#	5769#	5774#	5790#	5808#	5809#	5810#	5811#	5824#	5825#	5826#
5827#	5833#	5841#	5846#	5862#	5880#	5881#	5882#	5883#	5896#	5897#	5898#	5899#	5905#	5913#
5917#	5939#	5941#	5942#	5950#	5958#	5960#	5961#	5965#	5972#	5994#	5996#	5997#	6005#	6013#
6015#	6016#	6020#	6027#	6049#	6051#	6052#	6060#	6068#	6070#	6071#	6075#	6081#	6109#	6111#
6112#	6121#	6129#	6131#	6132#	6136#	6143#	6166#	6168#	6169#	6181#	6183#	6184#	6192#	6194#
6195#	6200#	6202#	6203#	6212#	6214#	6215#	6222#	6223#	6224#	6225#	6237#	6239#	6240#	6248#
6250#	6251#	6258#	6259#	6260#	6261#	6265#	6289#	6291#	6292#	6303#	6305#	6306#	6311#	6313#
6314#	6326#	6327#	6328#	6329#	6333#	6354#	6356#	6357#	6368#	6370#	6371#	6376#	6378#	6379#
6391#	6392#	6393#	6394#	6398#	6419#	6421#	6422#	6436#	6438#	6439#	6444#	6446#	6447#	6458#
6459#	6460#	6461#	6465#	6486#	6488#	6489#	6500#	6502#	6503#	6508#	6510#	6511#	6523#	6524#
6525#	6526#	6530#	6551#	6553#	6554#	6565#	6567#	6568#	6573#	6575#	6576#	6588#	6589#	6590#
6591#	6595#	6616#	6618#	6619#	6632#	6634#	6635#	6640#	6642#	6643#	6655#	6656#	6657#	6658#
6662#	6719#	6725#	6727#	6728#	6734#	6736#	6737#	6741#	6742#	6749#	6751#	6752#	6758#	6760#
6761#	6767#	6769#	6770#	6776#	6778#	6779#	6785#	6787#	6788#	6791#	6792#	6793#	6794#	6796#
6797#	6798	6799#	6800	6810#	6812#	6813#	6817#	6818#	6825#	6827#	6828#	6834#	6836#	6837#
6843#	6845#	6846#	6849#	6850#	6851#	6852#	6854#	6855#	6864#	6866#	6867#	6874#	6876#	6877#
6884#	6886#	6887#	6897#	6898#	6902#	6903#	6904#	6905#	6911#	6913#	6914#	6923#	6925#	6926#
6934#	6936#	6937#	6943#	6945#	6946#	6949#	6950#	6951#	6952#	6959#	6961#	6962#	6971#	6973#
6974#	6980#	6982#	6983#	6989#	6991#	6992#	6995#	6996#	6997#	6998#	7000#	7001#	7010#	7012#
7013#	7020#	7022#	7023#	7032#	7033#	7040#	7042#	7043#	7049#	7051#	7052#	7055#	7056#	7057#
7058#	7060#	7061#	7065#	7066#	7073#	7075#	7076#	7082#	7084#	7085#	7088#	7089#	7090#	7091#
7097#	7099#	7100#	7104#	7105#	7112#	7114#	7115#	7121#	7123#	7124#	7127#	7128#	7129#	7130#
7138#	7140#	7141#	7149#	7151#	7152#	7155#	7156#	7157#	7158#	7164#	7165#	7172#	7174#	7175#
7178#	7179#	7180#	7181#	7189#	7191#	7192#	7197#	7198#	7199#	7200#	7206#	7207#	7211#	7212#
7213#	7214#	7223#	7225#	7226#	7230#	7231#	7238#	7240#	7241#	7247#	7249#	7250#	7253#	7254#
7255#	7256#	7265#	7267#	7268#	7272#	7273#	7280#	7282#	7283#	7289#	7291#	7292#	7295#	7296#
7297#	7298#	7300#	7301#	7311#	7313#	7314#	7318#	7319#	7326#	7328#	7329#	7335#	7337#	7338#
7341#	7342#	7343#	7344#	7346#	7347#	7357#	7359#	7360#	7364#	7365#	7372#	7374#	7375#	7381#
7383#	7384#	7390#	7392#	7393#	7399#	7401#	7402#	7405#	7406#	7407#	7408#	7412#	7419#	7423#
7425#	7426#	7432#	7434#	7435#	7442#	7443#	7451#	7453#	7454#	7460#	7462#	7463#	7469#	7471#
7472#	7478#	7480#	7481#	7487#	7489#	7490#	7493#	7494#	7495#	7496#	7506#	7508#	7509#	7513#
7514#	7522#	7524#	7525#	7531#	7533#	7534#	7537#	7538#	7539#	7540#	7542#	7543#	7554#	7556#
7557#	7563#	7565#	7566#	7572#	7574#	7575#	7585#	7586#	7593#	7595#	7596#	7602#	7604#	7605#
7611#	7613#	7614#	7620#	7622#	7623#	7629#	7631#	7632#	7635#	7636#	7637#	7638#	7642#	7644#
7645#	7653#	7655#	7656#	7661#	7662#	7663#	7664#	7668#	7670#	7671#	7678#	7680#	7681#	7690#
7692#	7693#	7699#	7701#	7702#	7705#	7706#	7707#	7708#	7710#	7711#	7720#	7722#	7723#	7732#
7733#	7740#	7742#	7743#	7749#	7751#	7752#	7758#	7760#	7761#	7767#	7769#	7770#	7773#	7774#
7775#	7776#	7778#	7779#	7783#	7784#	7791#	7793#	7794#	7800#	7802#	7803#	7809#	7811#	7812#
7818#	7820#	7821#	7824#	7825#	7826#	7827#	7834#	7836#	7837#	7841#	7842#	7850#	7852#	7853#
7859#	7861#	7862#	7868#	7870#	7871#	7874#	7875#	7876#	7877#	7885#	7887#	7888#	7897#	7899#
7900#	7903#	7904#	7905#	7906#	7912#	7913#	7921#	7923#	7924#	7927#	7928#	7929#	7930#	7938#
7940#	7941#	7946#	7947#	7948#	7949#	7955#	7956#	7960#	7961#	7962#	7963#	7972#	7974#	7975#
7979#	7980#	7984#	7985#	7986#	7987#	7997#	7999#	8000#	8006#	8008#	8009#	8013#	8014#	8021#
8023#	8024#	8030#	8032#	8033#	8036#	8037#	8038#	8039#	8041#	8042#	8049#	8050#	8054#	8055#
8056#	8057#	8064#	8066#	8067#	8071#	8072#	8079#	8081#	8082#	8085#	8086#	8087#	8088#	8090#
8091#	8101#	8103#	8104#	8108#	8109#	8116#	8118#	8119#	8125#	8127#	8128#	8131#	8132#	8133#
8134#	8141#	8143#	8144#	8148#	8149#	8156#	8158#	8159#	8165#	8167#	8168#	8174#	8176#	8177#
8183#	8185#	8186#	8192#	8194#	8195#	8198#	8199#	8200#	8201#	8204#	8207#	8242#	8281	8282
8283	8284	8306#	8369#	8373#	8375#	8376#	8382#	8384#	8385#	8389#	8390#	8394#	8395#	8396#
8397#	8399#	8400#	8413#	8414#	8419#	8420#	8421#	8422#	8424#	8425#	8432#	8434#	8435#	8439#
8440#	8444#	8445#	8446#	8447#	8451#	8453#	8454#	8461#	8462#	8471#	8473#	8474#	8478#	8479#
8483#	8484#	8485#	8486#	8491#	8492#	8497#	8498#	8499#	8500#	8502#	8503#	8510#	8512#	8513#



CVDMAA.P11 12-DEC-80 15:59

CROSS REFERENCE TABLE -- MACRO NAMES

	8517#	8518#	8522#	8523#	8524#	8525#	8527#	8528#	8535#	8537#	8538#	8542#	8543#	8547#	8548#
	8549#	8550#	8554#	8558#	8562#	8564#	8565#	8572#	8574#	8575#	8582#	8584#	8585#	8591#	8593#
	8594#	8598#	8599#	8603#	8604#	8605#	8606#	8608#	8609#	8622#	8623#	8628#	8629#	8630#	8631#
	8633#	8634#	8642#	8643#	8647#	8648#	8649#	8650#	8652#	8653#	8662#	8664#	8665#	8669#	8670#
	8674#	8675#	8676#	8677#	8679#	8680#	8688#	8689#	8694#	8695#	8696#	8697#	8699#	8700#	8708#
	8709#	8713#	8714#	8715#	8716#	8723#	8725#	8726#	8733#	8734#	8743#	8745#	8746#	8750#	8751#
	8755#	8756#	8757#	8758#	8760#	8761#	8766#	8767#	8772#	8773#	8774#	8775#	8777#	8778#	8785#
	8787#	8788#	8792#	8793#	8797#	8798#	8799#	8800#	8802#	8803#	8807#	8808#	8815#	8817#	8818#
	8824#	8826#	8827#	8833#	8835#	8836#	8842#	8844#	8845#	8851#	8853#	8854#	8857#	8858#	8859#
	8860#	8872#	8874#	8875#	8879#	8880#	8884#	8885#	8886#	8887#	8889#	8890#	8896#	8900#	8903#
	8920#	8924#	8925	8926	8927	8929#	8930	8931	8932	8934#	8935	8936	8937	8938	8941#
	8960#	8963#	8977#	8978#	8979#										
MSGNLS	1#	1242#	5760#	5832#	5904#										
MSGNSU	1#	1242#	4266#	4319#	5371#	5426#	5492#	5613#	5773#	5845#	5949#	6004#	6059#	6120#	6718#
	7418#	8368#	8557#												
MSGNTA	1#	1242#	1428#	1441#	2599#	2633#	2651#	2682#	2692#	2732#	2759#	2966#	2998#	3056#	3159#
	3321#	3635#	3695#	3715#	3730#	3743#	3807#	3849#	3912#	4017#	4100#	4159#	4236#	4315#	4387#
	4390#	4538#	4615#	5105#	5339#	5386#	5393#	5441#	5448#	5507#	5514#	5768#	5840#	5912#	5916#
	5964#	5971#	6019#	6026#	6074#	6080#	6135#	6142#	6264#	6332#	6397#	6464#	6529#	6594#	6661#
	7411#	8203#	8206#	8553#	8899#	8902#	8941#	8942	8963#	8964					
MSGNTE	1#	1242#	3767#	3875#	3930#	4037#	4114#	4174#	4264#	4411#	4559#	4861#	5139#	5357#	5412#
	5468#	5581#	5935#	5990#	6045#	6105#	6161#	6284#	6349#	6414#	6481#	6546#	6611#	6716#	8366#
MSHAPT	1#	1242#	1271#												
MSHNAP	1#	1242#	1271#	1310											
MSINCR	1#	1242#	1244#	1406#	1437#	2580#	2587#	2614#	2621#	2648#	2652#	2657#	2664#	2677#	2683#
	2688#	2693#	2698#	2709#	2726#	2733#	2738#	2745#	2755#	2760#	2765#	2774#	2793#	2803#	2823#
	2833#	2853#	2863#	2872#	2891#	2901#	2921#	2931#	2951#	2961#	2967#	2972#	2981#	2994#	2999#
	3004#	3019#	3027#	3035#	3043#	3051#	3057#	3082#	3090#	3106#	3113#	3121#	3129#	3138#	3146#
	3154#	3160#	3195#	3216#	3224#	3236#	3253#	3261#	3278#	3288#	3305#	3315#	3322#	3355#	3369#
	3376#	3390#	3397#	3411#	3418#	3432#	3447#	3457#	3464#	3474#	3486#	3503#	3516#	3522#	3528#
	3534#	3540#	3557#	3562#	3580#	3620#	3628#	3636#	3662#	3668#	3687#	3692#	3696#	3707#	3710#
	3713#	3716#	3725#	3728#	3731#	3741#	3744#	3767#	3768#	3776#	3795#	3802#	3808#	3820#	3826#
	3832#	3838#	3846#	3850#	3875#	3876#	3882#	3907#	3913#	3930#	3931#	4012#	4018#	4037#	4038#
	4044#	4085#	4101#	4114#	4115#	4123#	4139#	4153#	4160#	4174#	4175#	4186#	4194#	4217#	4230#
	4237#	4264#	4265#	4266#	4267#	4271#	4310#	4316#	4319#	4320#	4332#	4337#	4382#	4388#	4391#
	4411#	4412#	4416#	4426#	4435#	4440#	4494#	4499#	4511#	4516#	4527#	4532#	4539#	4559#	4560#
	4563#	4565#	4578#	4580#	4586#	4596#	4601#	4613#	4616#	4861#	4862#	4865#	4867#	4892#	4894#
	4911#	4913#	4933#	4940#	4942#	4956#	4972#	4974#	4985#	4987#	5001#	5103#	5106#	5139#	5140#
	5143#	5151#	5158#	5176#	5191#	5219#	5224#	5234#	5241#	5248#	5256#	5261#	5270#	5277#	5282#
	5290#	5297#	5309#	5314#	5322#	5330#	5335#	5340#	5357#	5358#	5361#	5363#	5371#	5372#	5380#
	5382#	5387#	5394#	5412#	5413#	5416#	5418#	5426#	5427#	5435#	5437#	5442#	5449#	5468#	5469#
	5472#	5474#	5482#	5484#	5492#	5493#	5501#	5503#	5508#	5515#	5581#	5582#	5588#	5590#	5601#
	5603#	5613#	5614#	5623#	5642#	5663#	5686#	5712#	5729#	5752#	5761#	5769#	5773#	5774#	5790#
	5808#	5824#	5833#	5841#	5845#	5846#	5862#	5880#	5896#	5905#	5913#	5917#	5935#	5936#	5939#
	5941#	5949#	5950#	5958#	5960#	5965#	5972#	5990#	5991#	5994#	5996#	6004#	6005#	6013#	6015#
	6020#	6027#	6045#	6046#	6049#	6051#	6059#	6060#	6068#	6070#	6075#	6081#	6105#	6106#	6109#
	6111#	6120#	6121#	6129#	6131#	6136#	6143#	6161#	6162#	6166#	6168#	6181#	6183#	6192#	6194#
	6200#	6202#	6212#	6214#	6222#	6237#	6239#	6248#	6250#	6258#	6265#	6284#	6285#	6289#	6291#
	6303#	6305#	6311#	6313#	6326#	6333#	6349#	6350#	6354#	6356#	6368#	6370#	6376#	6378#	6391#
	6398#	6414#	6415#	6419#	6421#	6436#	6438#	6444#	6446#	6458#	6465#	6481#	6482#	6486#	6488#
	6500#	6502#	6508#	6510#	6523#	6530#	6546#	6547#	6551#	6553#	6565#	6567#	6573#	6575#	6588#
	6595#	6611#	6612#	6616#	6618#	6632#	6634#	6640#	6642#	6655#	6662#	6716#	6717#	6718#	6719#
	6725#	6727#	6734#	6736#	6741#	6749#	6751#	6758#	6760#	6767#	6769#	6776#	6778#	6785#	6787#
	6791#	6799#	6810#	6812#	6817#	6825#	6827#	6834#	6836#	6843#	6845#	6849#	6854#	6864#	6866#
	6874#	6876#	6884#	6886#	6897#	6902#	6911#	6913#	6923#	6925#	6934#	6936#	6943#	6945#	6949#
	6959#	6961#	6971#	6973#	6980#	6982#	6989#	6991#	6995#	7000#	7010#	7012#	7020#	7022#	7032#



CVDMAA.P11 12-DEC-80 15:59

CROSS REFERENCE TABLE -- MACRO NAMES

	7040#	7042#	7049#	7051#	7055#	7060#	7065#	7073#	7075#	7082#	7084#	7088#	7097#	7099#	7104#
	7112#	7114#	7121#	7123#	7127#	7138#	7140#	7149#	7151#	7155#	7164#	7172#	7174#	7178#	7189#
	7191#	7197#	7206#	7211#	7223#	7225#	7230#	7238#	7240#	7247#	7249#	7253#	7265#	7267#	7272#
	7280#	7282#	7289#	7291#	7295#	7300#	7311#	7313#	7318#	7326#	7328#	7335#	7337#	7341#	7346#
	7357#	7359#	7364#	7372#	7374#	7381#	7383#	7390#	7392#	7399#	7401#	7405#	7412#	7418#	7419#
	7423#	7425#	7432#	7434#	7442#	7451#	7453#	7460#	7462#	7469#	7471#	7478#	7480#	7487#	7489#
	7493#	7506#	7508#	7513#	7522#	7524#	7531#	7533#	7537#	7542#	7554#	7556#	7563#	7565#	7572#
	7574#	7585#	7593#	7595#	7602#	7604#	7611#	7613#	7620#	7622#	7629#	7631#	7635#	7642#	7644#
	7653#	7655#	7661#	7668#	7670#	7678#	7680#	7690#	7692#	7699#	7701#	7705#	7710#	7720#	7722#
	7732#	7740#	7742#	7749#	7751#	7758#	7760#	7767#	7769#	7773#	7778#	7783#	7791#	7793#	7800#
	7802#	7809#	7811#	7818#	7820#	7824#	7834#	7836#	7841#	7850#	7852#	7859#	7861#	7868#	7870#
	7874#	7885#	7887#	7897#	7899#	7903#	7912#	7921#	7923#	7927#	7938#	7940#	7946#	7955#	7960#
	7972#	7974#	7979#	7984#	7997#	7999#	8006#	8008#	8013#	8021#	8023#	8030#	8032#	8036#	8041#
	8049#	8054#	8064#	8066#	8071#	8079#	8081#	8085#	8090#	8101#	8103#	8108#	8116#	8118#	8125#
	8127#	8131#	8141#	8143#	8148#	8156#	8158#	8165#	8167#	8174#	8176#	8183#	8185#	8192#	8194#
	8198#	8204#	8207#	8242#	8306#	8366#	8367#	8368#	8369#	8373#	8375#	8382#	8384#	8389#	8394#
	8399#	8413#	8419#	8424#	8432#	8434#	8439#	8444#	8451#	8453#	8461#	8471#	8473#	8478#	8483#
	8491#	8497#	8502#	8510#	8512#	8517#	8522#	8527#	8535#	8537#	8542#	8547#	8554#	8557#	8558#
	8562#	8564#	8572#	8574#	8582#	8584#	8591#	8593#	8598#	8603#	8608#	8622#	8628#	8633#	8642#
	8647#	8652#	8662#	8664#	8669#	8674#	8679#	8688#	8694#	8699#	8708#	8713#	8723#	8725#	8733#
	8743#	8745#	8750#	8755#	8760#	8766#	8772#	8777#	8785#	8787#	8792#	8797#	8802#	8807#	8815#
	8817#	8824#	8826#	8833#	8835#	8842#	8844#	8851#	8853#	8857#	8872#	8874#	8879#	8884#	8889#
	8896#	8900#	8903#	8920#	8960#										
MSIOSE	1#	1242#													
MSLDRO	1#	1242#	3521#	3527#	3533#	3539#	3561#	3579#	3686#	3691#	3709#	3712#	3794#		
MSMASK	1#	1242#													
MSPCHI	1#	1242#													
MSPCLO	1#	1242#													
MSPSK1	1#	1242#													
MSPPOP	1#	1242#	1428#	1441#	2599#	2633#	2651#	2682#	2692#	2732#	2759#	2966#	2998#	3056#	3159#
	3321#	3508#	3635#	3695#	3715#	3730#	3743#	3807#	3849#	3912#	4017#	4100#	4159#	4236#	4315#
	4387#	4390#	4538#	4615#	5105#	5339#	5386#	5393#	5441#	5448#	5507#	5514#	5760#	5768#	5832#
	5840#	5904#	5912#	5916#	5964#	5971#	6019#	6026#	6074#	6080#	6135#	6142#	6264#	6332#	6397#
	6464#	6529#	6594#	6661#	7411#	8203#	8206#	8553#	8899#	8902#	8941#	8963#	8976#		
MSPRIN	1#	1242#	2659#	2668#	2704#	2720#	2740#	2749#	2770#	2777#	2796#	2807#	2826#	2837#	2856#
	2868#	2875#	2894#	2905#	2924#	2935#	2954#	2976#	2985#	3015#	3024#	3032#	3040#	3048#	3072#
	3087#	3110#	3118#	3126#	3135#	3143#	3151#	3184#	3208#	3221#	3249#	3258#	3266#	3281#	3293#
	3308#	3350#	3358#	3372#	3379#	3393#	3400#	3414#	3421#	3442#	3450#	3460#	3467#	3483#	3822#
	3829#	3835#	3841#	6796#											
MSPUSH	1#	1242#	1244#	1406#	1437#	2580#	2614#	2648#	2657#	2688#	2698#	2738#	2765#	2972#	3004#
	3106#	3236#	3503#	3516#	3662#	3707#	3725#	3741#	3767#	3768	3820#	3875#	3876	3930#	3931
	4037#	4038	4114#	4115	4174#	4175	4264#	4265	4266#	4267	4319#	4320	4411#	4412	4559#
	4560	4861#	4862	5139#	5140	5357#	5358	5371#	5372	5412#	5413	5426#	5427	5468#	5469
	5492#	5493	5581#	5582	5613#	5614	5623#	5773#	5774	5790#	5845#	5846	5862#	5935#	5936
	5949#	5950	5990#	5991	6004#	6005	6045#	6046	6059#	6060	6105#	6106	6120#	6121	6161#
	6162	6284#	6285	6349#	6350	6414#	6415	6481#	6482	6546#	6547	6611#	6612	6716#	6717
	6718#	6719	7418#	7419	8366#	8367	8368#	8369	8557#	8558	8920#	8960#			
MSPUT	1#	1242#	2659#	2668#	2704#	2720#	2740#	2749#	2770#	2777#	2796#	2807#	2826#	2837#	2856#
	2868#	2875#	2894#	2905#	2924#	2935#	2954#	2976#	2985#	3015#	3024#	3032#	3040#	3048#	3072#
	3087#	3110#	3118#	3126#	3135#	3143#	3151#	3184#	3208#	3221#	3249#	3258#	3266#	3281#	3293#
	3308#	3350#	3358#	3372#	3379#	3393#	3400#	3414#	3421#	3442#	3450#	3460#	3467#	3483#	3553#
	3616#	3624#	3664#	3772#	3822#	3829#	3835#	3841#	6796#						
MSPUT1	1#	1242#	2659#	2660	2661	2662	2668#	2670	2672	2674	2675	2704#	2706	2707	2720#
	2721	2723	2724	2740#	2741	2742	2743	2749#	2750	2751	2752	2753	2770#	2771	2772
	2777#	2779	2781	2783	2785	2787	2789	2790	2791	2796#	2798	2800	2801	2807#	2809
	2811	2813	2815	2817	2819	2820	2821	2826#	2828	2830	2831	2837#	2839	2841	2843



CVDMAA.P11 12-DEC-80 15:59

CROSS REFERENCE TABLE -- MACRO NAMES

	2845	2847	2849	2850	2851	2856#	2858	2860	2861	2868#	2869	2870	2875#	2877	2879
	2881	2883	2885	2887	2888	2889	2894#	2896	2898	2899	2905#	2907	2909	2911	2913
	2915	2917	2918	2919	2924#	2926	2928	2929	2935#	2937	2939	2941	2943	2945	2947
	2948	2949	2954#	2956	2958	2959	2976#	2977	2978	2979	2985#	2987	2989	2991	2992
	3015#	3016	3017	3024#	3025	3032#	3033	3040#	3041	3048#	3049	3072#	3074	3076	3078
	3079	3080	3087#	3088	3110#	3111	3118#	3119	3126#	3127	3135#	3136	3143#	3144	3151#
	3152	3184#	3186	3188	3190	3191	3192	3193	3208#	3210	3211	3212	3213	3214	3221#
	3222	3249#	3250	3251	3258#	3259	3266#	3268	3270	3272	3274	3275	3276	3281#	3283
	3285	3286	3293#	3295	3297	3299	3301	3302	3303	3308#	3310	3312	3313	3350#	3351
	3352	3353	3358#	3360	3362	3364	3366	3367	3372#	3373	3374	3379#	3381	3383	3385
	3387	3388	3393#	3394	3395	3400#	3402	3404	3406	3408	3409	3414#	3415	3416	3421#
	3423	3425	3427	3429	3430	3442#	3443	3444	3445	3450#	3451	3452	3453	3454	3455
	3460#	3461	3462	3467#	3468	3469	3470	3471	3472	3483#	3484	3553#	3554	3555	3556
	3616#	3617	3618	3619	3624#	3625	3626	3627	3664#	3665	3666	3667	3772#	3773	3774
	3775	3822#	3823	3824	3829#	3830	3835#	3836	3841#	3842	3843	3844	6796#	6797	
MSRADI	1#	1242#	8924#	8929#	8934#										
MSRBRO	1#	1242#													
MSRNRO	1#	1242#	3579#	3581											
MSSETS	1#	1242#	1244#	1406#	1437#	2580#	2614#	2648#	2657#	2688#	2698#	2738#	2765#	2972#	3004#
	3106#	3236#	3503#	3516#	3662#	3707#	3725#	3741#	3768#	3820#	3876#	3931#	4038#	4115#	4175#
	4265#	4267#	4320#	4412#	4560#	4862#	5140#	5358#	5372#	5413#	5427#	5469#	5493#	5582#	5614#
	5623#	5774#	5790#	5846#	5862#	5936#	5950#	5991#	6005#	6046#	6060#	6106#	6121#	6162#	6285#
	6350#	6415#	6482#	6547#	6612#	6717#	6719#	7419#	8367#	8369#	8558#	8920#	8960#		
MSSTAR	1#	1242#													
MS SVC	1#	1242#	2587	2621	2651#	2652	2659#	2664	2668#	2677	2682#	2683	2692#	2693	2704#
	2709	2720#	2726	2732#	2733	2740#	2745	2749#	2755	2759#	2760	2770#	2774	2777#	2793
	2796#	2803	2807#	2823	2826#	2833	2837#	2853	2856#	2863	2868#	2872	2875#	2891	2894#
	2901	2905#	2921	2924#	2931	2935#	2951	2954#	2961	2966#	2967	2976#	2981	2985#	2994
	2998#	2999	3015#	3019	3024#	3027	3032#	3035	3040#	3043	3048#	3051	3056#	3057	3072#
	3082	3087#	3090	3110#	3113	3118#	3121	3126#	3129	3135#	3138	3143#	3146	3151#	3154
	3159#	3160	3184#	3195	3208#	3216	3221#	3224	3249#	3253	3258#	3261	3266#	3278	3281#
	3288	3293#	3305	3308#	3315	3321#	3322	3350#	3355	3358#	3369	3372#	3376	3379#	3390
	3393#	3397	3400#	3411	3414#	3418	3421#	3432	3442#	3447	3450#	3457	3460#	3464	3467#
	3474	3483#	3486	3521#	3522	3527#	3528	3533#	3534	3539#	3540	3553#	3557	3561#	3562
	3579#	3580	3616#	3620	3624#	3628	3635#	3636	3664#	3668	3686#	3687	3691#	3692	3695#
	3696	3709#	3710	3712#	3713	3715#	3716	3728#	3730#	3731	3743#	3744	3772#	3776	3794#
	3795	3802	3807#	3808	3822#	3826	3829#	3832	3835#	3838	3841#	3846	3849#	3850	3882#
	3907	3912#	3913	4012	4017#	4018	4044#	4085	4100#	4101	4123#	4139#	4153	4159#	4160
	4186#	4194#	4217	4230	4236#	4237	4266#	4267	4271#	4310	4315#	4316	4319#	4320	4332
	4337#	4382	4387#	4388	4390#	4391	4416#	4426#	4435	4440#	4494	4499#	4511	4516#	4527
	4532#	4538#	4539	4563#	4565#	4578#	4580#	4586#	4596#	4601#	4613#	4615#	4616	4865#	4867#
	4892#	4894#	4911#	4913#	4933#	4940#	4942#	4956	4972#	4974#	4985#	4987#	5001	5103#	5105#
	5106	5143#	5151#	5158#	5176#	5191#	5219	5224#	5234#	5241#	5248#	5256	5261#	5270#	5277
	5282#	5290#	5297#	5309	5314#	5322#	5330	5335#	5339#	5340	5361#	5363#	5371#	5372	5380#
	5382#	5386#	5387	5393#	5394	5416#	5418#	5426#	5427	5435#	5437#	5441#	5442	5448#	5449
	5472#	5474#	5482#	5484#	5492#	5493	5501#	5503#	5507#	5508	5514#	5515	5588#	5590#	5601#
	5603#	5613#	5614	5623#	5642	5663	5686	5712	5729	5752	5760#	5761	5768#	5769	5773#
	5774	5790#	5808	5824	5832#	5833	5840#	5841	5845#	5846	5862#	5880	5896	5904#	5905
	5912#	5913	5916#	5917	5939#	5941#	5949#	5950	5958#	5960#	5964#	5965	5971#	5972	5994#
	5996#	6004#	6005	6013#	6015#	6019#	6020	6026#	6027	6049#	6051#	6059#	6060	6068#	6070#
	6074#	6075	6080#	6081	6109#	6111#	6120#	6121	6129#	6131#	6135#	6136	6142#	6143	6166#
	6168#	6181#	6183#	6192#	6194#	6200#	6202#	6212#	6214#	6222	6237#	6239#	6248#	6250#	6258
	6264#	6265	6289#	6291#	6303#	6305#	6311#	6313#	6326	6332#	6333	6354#	6356#	6368#	6370#
	6376#	6378#	6391	6397#	6398	6419#	6421#	6436#	6438#	6444#	6446#	6458	6464#	6465	6486#
	6488#	6500#	6502#	6508#	6510#	6523	6529#	6530	6551#	6553#	6565#	6567#	6573#	6575#	6588
	6594#	6595	6616#	6618#	6632#	6634#	6640#	6642#	6655	6661#	6662	6718#	6719	6725#	6727#



CVDMAA.P11

12-DEC-80 15:59

## CROSS REFERENCE TABLE -- MACRO NAMES

MSTLAB

6734#	6736#	6741#	6749#	6751#	6758#	6760#	6767#	6769#	6776#	6778#	6785#	6787#	6791	6796#
6799	6810#	6812#	6817#	6825#	6827#	6834#	6836#	6843#	6845#	6849	6854#	6864#	6866#	6874#
6876#	6884#	6886#	6897#	6902	6911#	6913#	6923#	6925#	6934#	6936#	6943#	6945#	6949	6959#
6961#	6971#	6973#	6980#	6982#	6989#	6991#	6995	7000#	7010#	7012#	7020#	7022#	7032#	7040#
7042#	7049#	7051#	7055	7060#	7065#	7073#	7075#	7082#	7084#	7088	7097#	7099#	7104#	7112#
7114#	7121#	7123#	7127	7138#	7140#	7149#	7151#	7155	7164#	7172#	7174#	7178	7189#	7191#
7197	7206#	7211	7223#	7225#	7230#	7238#	7240#	7247#	7249#	7253	7265#	7267#	7272#	7280#
7282#	7289#	7291#	7295	7300#	7311#	7313#	7318#	7326#	7328#	7335#	7337#	7341	7346#	7357#
7359#	7364#	7372#	7374#	7381#	7383#	7390#	7392#	7399#	7401#	7405	7411#	7412	7418#	7419
7423#	7425#	7432#	7434#	7442#	7451#	7453#	7460#	7462#	7469#	7471#	7478#	7480#	7487#	7489#
7493	7506#	7508#	7513#	7522#	7524#	7531#	7533#	7537	7542#	7554#	7556#	7563#	7565#	7572#
7574#	7585#	7593#	7595#	7602#	7604#	7611#	7613#	7620#	7622#	7629#	7631#	7635	7642#	7644#
7653#	7655#	7661	7668#	7670#	7678#	7680#	7690#	7692#	7699#	7701#	7705	7710#	7720#	7722#
7732#	7740#	7742#	7749#	7751#	7758#	7760#	7767#	7769#	7773	7778#	7783#	7791#	7793#	7800#
7802#	7809#	7811#	7818#	7820#	7824	7834#	7836#	7841#	7850#	7852#	7859#	7861#	7868#	7870#
7874	7885#	7887#	7897#	7899#	7903	7912#	7921#	7923#	7927	7938#	7940#	7946	7955#	7960
7972#	7974#	7979#	7984	7997#	7999#	8006#	8008#	8013#	8021#	8023#	8030#	8032#	8036	8041#
8049#	8054	8064#	8066#	8071#	8079#	8081#	8085	8090#	8101#	8103#	8108#	8116#	8118#	8125#
8127#	8131	8141#	8143#	8148#	8156#	8158#	8165#	8167#	8174#	8176#	8183#	8185#	8192#	8194#
8198	8203#	8204	8206#	8207	8242#	8306#	8368#	8369	8373#	8375#	8382#	8384#	8389#	8394
8399#	8413#	8419	8424#	8432#	8434#	8439#	8444	8451#	8453#	8461#	8471#	8473#	8478#	8483
8491#	8497	8502#	8510#	8512#	8517#	8522	8527#	8535#	8537#	8542#	8547	8553#	8554	8557#
8558	8562#	8564#	8572#	8574#	8582#	8584#	8591#	8593#	8598#	8603	8608#	8622#	8628	8633#
8642#	8647	8652#	8662#	8664#	8669#	8674	8679#	8688#	8694	8699#	8708#	8713	8723#	8725#
8733#	8743#	8745#	8750#	8755	8760#	8766#	8772	8777#	8785#	8787#	8792#	8797	8802#	8807#
8815#	8817#	8824#	8826#	8833#	8835#	8842#	8844#	8851#	8853#	8857	8872#	8874#	8879#	8884
8889#	8896#	8899#	8900	8902#	8903									
1#	1242#	2587#	2621#	2652#	2664#	2677#	2683#	2693#	2709#	2726#	2733#	2745#	2755#	2760#
2774#	2793#	2803#	2823#	2833#	2853#	2863#	2872#	2891#	2901#	2921#	2931#	2951#	2961#	2967#
2981#	2994#	2999#	3019#	3027#	3035#	3043#	3051#	3057#	3082#	3090#	3113#	3121#	3129#	3138#
3146#	3154#	3160#	3195#	3216#	3224#	3253#	3261#	3278#	3288#	3305#	3315#	3322#	3355#	3369#
3376#	3390#	3397#	3411#	3418#	3432#	3447#	3457#	3464#	3474#	3486#	3522#	3528#	3534#	3540#
3557#	3562#	3580#	3620#	3628#	3636#	3668#	3687#	3692#	3696#	3710#	3713#	3716#	3728#	3731#
3744#	3776#	3795#	3802#	3808#	3826#	3832#	3838#	3846#	3850#	3882#	3907#	3913#	4012#	4018#
4044#	4085#	4101#	4123#	4139#	4153#	4160#	4186#	4194#	4217#	4230#	4237#	4267#	4271#	4310#
4316#	4320#	4332#	4337#	4382#	4388#	4391#	4416#	4426#	4435#	4440#	4494#	4499#	4511#	4516#
4527#	4532#	4539#	4563#	4565#	4578#	4580#	4586#	4596#	4601#	4613#	4616#	4865#	4867#	4892#
4894#	4911#	4913#	4933#	4940#	4942#	4956#	4972#	4974#	4985#	4987#	5001#	5103#	5106#	5143#
5151#	5158#	5176#	5191#	5219#	5224#	5234#	5241#	5248#	5256#	5261#	5270#	5277#	5282#	5290#
5297#	5309#	5314#	5322#	5330#	5335#	5340#	5361#	5363#	5372#	5380#	5382#	5387#	5394#	5416#
5418#	5427#	5435#	5437#	5442#	5449#	5472#	5474#	5482#	5484#	5493#	5501#	5503#	5508#	5515#
5588#	5590#	5601#	5603#	5614#	5623#	5642#	5663#	5686#	5712#	5729#	5752#	5761#	5769#	5774#
5790#	5808#	5824#	5833#	5841#	5846#	5862#	5880#	5896#	5905#	5913#	5917#	5939#	5941#	5950#
5958#	5960#	5965#	5972#	5994#	5996#	6005#	6013#	6015#	6020#	6027#	6049#	6051#	6060#	6068#
6070#	6075#	6081#	6109#	6111#	6121#	6129#	6131#	6136#	6143#	6166#	6168#	6181#	6183#	6192#
6194#	6200#	6202#	6212#	6214#	6222#	6237#	6239#	6248#	6250#	6258#	6265#	6289#	6291#	6303#
6305#	6311#	6313#	6326#	6333#	6354#	6356#	6368#	6370#	6376#	6378#	6391#	6398#	6419#	6421#
6436#	6438#	6444#	6446#	6458#	6465#	6486#	6488#	6500#	6502#	6508#	6510#	6523#	6530#	6551#
6553#	6565#	6567#	6573#	6575#	6588#	6595#	6616#	6618#	6632#	6634#	6640#	6642#	6655#	6662#
6719#	6725#	6727#	6734#	6736#	6741#	6749#	6751#	6758#	6760#	6767#	6769#	6776#	6778#	6785#
6787#	6791#	6799#	6810#	6812#	6817#	6825#	6827#	6834#	6836#	6843#	6845#	6849#	6854#	6864#
6866#	6874#	6876#	6884#	6886#	6897#	6902#	6911#	6913#	6923#	6925#	6934#	6936#	6943#	6945#
6949#	6959#	6961#	6971#	6973#	6980#	6982#	6989#	6991#	6995#	7000#	7010#	7012#	7020#	7022#
7032#	7040#	7042#	7049#	7051#	7055#	7060#	7065#	7073#	7075#	7082#	7084#	7088#	7097#	7099#
7104#	7112#	7114#	7121#	7123#	7127#	7138#	7140#	7149#	7151#	7155#	7164#	7172#	7174#	7178#
7189#	7191#	7197#	7206#	7211#	7223#	7225#	7230#	7238#	7240#	7247#	7249#	7253#	7265#	7267#



CVDMAA.P11 12-DEC-80 15:59

CROSS REFERENCE TABLE -- MACRO NAMES

MSTSTL

7272#	7280#	7282#	7289#	7291#	7295#	7300#	7311#	7313#	7318#	7326#	7328#	7335#	7337#	7341#
7346#	7357#	7359#	7364#	7372#	7374#	7381#	7383#	7390#	7392#	7399#	7401#	7405#	7412#	7419#
7423#	7425#	7432#	7434#	7442#	7451#	7453#	7460#	7462#	7469#	7471#	7478#	7480#	7487#	7489#
7493#	7506#	7508#	7513#	7522#	7524#	7531#	7533#	7537#	7542#	7554#	7556#	7563#	7565#	7572#
7574#	7585#	7593#	7595#	7602#	7604#	7611#	7613#	7620#	7622#	7629#	7631#	7635#	7642#	7644#
7653#	7655#	7661#	7668#	7670#	7678#	7680#	7690#	7692#	7699#	7701#	7705#	7710#	7720#	7722#
7732#	7740#	7742#	7749#	7751#	7758#	7760#	7767#	7769#	7773#	7778#	7783#	7791#	7793#	7800#
7802#	7809#	7811#	7818#	7820#	7824#	7834#	7836#	7841#	7850#	7852#	7859#	7861#	7868#	7870#
7874#	7885#	7887#	7897#	7899#	7903#	7912#	7921#	7923#	7927#	7938#	7940#	7946#	7955#	7960#
7972#	7974#	7979#	7984#	7997#	7999#	8006#	8008#	8013#	8021#	8023#	8030#	8032#	8036#	8041#
8049#	8054#	8064#	8066#	8071#	8079#	8081#	8085#	8090#	8101#	8103#	8108#	8116#	8118#	8125#
8127#	8131#	8141#	8143#	8148#	8156#	8158#	8165#	8167#	8174#	8176#	8183#	8185#	8192#	8194#
8198#	8204#	8207#	8242#	8306#	8369#	8373#	8375#	8382#	8384#	8389#	8394#	8399#	8413#	8419#
8424#	8432#	8434#	8439#	8444#	8451#	8453#	8461#	8471#	8473#	8478#	8483#	8491#	8497#	8502#
8510#	8512#	8517#	8522#	8527#	8535#	8537#	8542#	8547#	8554#	8558#	8562#	8564#	8572#	8574#
8582#	8584#	8591#	8593#	8598#	8603#	8608#	8622#	8628#	8633#	8642#	8647#	8652#	8662#	8664#
8669#	8674#	8679#	8688#	8694#	8699#	8708#	8713#	8723#	8725#	8733#	8743#	8745#	8750#	8755#
8760#	8766#	8772#	8777#	8785#	8787#	8792#	8797#	8802#	8807#	8815#	8817#	8824#	8826#	8833#
8835#	8842#	8844#	8851#	8853#	8857#	8872#	8874#	8879#	8884#	8889#	8896#	8900#	8903#	
1#	1242#	2587#	2621#	2652#	2664#	2677#	2683#	2693#	2709#	2726#	2733#	2745#	2755#	2760#
2774#	2793#	2803#	2823#	2833#	2853#	2863#	2872#	2891#	2901#	2921#	2931#	2951#	2961#	2967#
2981#	2994#	2999#	3019#	3027#	3035#	3043#	3051#	3057#	3082#	3090#	3113#	3121#	3129#	3138#
3146#	3154#	3160#	3195#	3216#	3224#	3253#	3261#	3278#	3288#	3305#	3315#	3322#	3355#	3369#
3376#	3390#	3397#	3411#	3418#	3432#	3447#	3457#	3464#	3474#	3486#	3522#	3528#	3534#	3540#
3557#	3562#	3580#	3620#	3628#	3636#	3668#	3687#	3692#	3696#	3710#	3713#	3716#	3728#	3731#
3744#	3776#	3795#	3802#	3808#	3826#	3832#	3838#	3846#	3850#	3882#	3907#	3913#	4012#	4018#
4044#	4085#	4101#	4123#	4139#	4153#	4160#	4186#	4194#	4217#	4230#	4237#	4267#	4271#	4310#
4316#	4320#	4332#	4337#	4382#	4388#	4391#	4416#	4426#	4435#	4440#	4494#	4499#	4511#	4516#
4527#	4532#	4539#	4563#	4565#	4578#	4580#	4586#	4596#	4601#	4613#	4616#	4865#	4867#	4892#
4894#	4911#	4913#	4933#	4940#	4942#	4956#	4972#	4974#	4985#	4987#	5001#	5103#	5106#	5143#
5151#	5158#	5176#	5191#	5219#	5224#	5234#	5241#	5248#	5256#	5261#	5270#	5277#	5282#	5290#
5297#	5309#	5314#	5322#	5330#	5335#	5340#	5361#	5363#	5372#	5380#	5382#	5387#	5394#	5416#
5418#	5427#	5435#	5437#	5442#	5449#	5472#	5474#	5482#	5484#	5493#	5501#	5503#	5508#	5515#
5588#	5590#	5601#	5603#	5614#	5623#	5642#	5663#	5686#	5712#	5729#	5752#	5761#	5769#	5774#
5790#	5808#	5824#	5833#	5841#	5846#	5862#	5880#	5896#	5905#	5913#	5917#	5939#	5941#	5950#
5958#	5960#	5965#	5972#	5994#	5996#	6005#	6013#	6015#	6020#	6027#	6049#	6051#	6060#	6068#
6070#	6075#	6081#	6109#	6111#	6121#	6129#	6131#	6136#	6143#	6166#	6168#	6181#	6183#	6192#
6194#	6200#	6202#	6212#	6214#	6222#	6237#	6239#	6248#	6250#	6258#	6265#	6289#	6291#	6303#
6305#	6311#	6313#	6326#	6333#	6354#	6356#	6368#	6370#	6376#	6378#	6391#	6398#	6419#	6421#
6436#	6438#	6444#	6446#	6458#	6465#	6486#	6488#	6500#	6502#	6508#	6510#	6523#	6530#	6551#
6553#	6565#	6567#	6573#	6575#	6588#	6595#	6616#	6618#	6632#	6634#	6640#	6642#	6655#	6662#
6719#	6725#	6727#	6734#	6736#	6741#	6749#	6751#	6758#	6760#	6767#	6769#	6776#	6778#	6785#
6787#	6791#	6799#	6810#	6812#	6817#	6825#	6827#	6834#	6836#	6843#	6845#	6849#	6854#	6864#
6866#	6874#	6876#	6884#	6886#	6897#	6902#	6911#	6913#	6923#	6925#	6934#	6936#	6943#	6945#
6949#	6959#	6961#	6971#	6973#	6980#	6982#	6989#	6991#	6995#	7000#	7010#	7012#	7020#	7022#
7032#	7040#	7042#	7049#	7051#	7055#	7060#	7065#	7073#	7075#	7082#	7084#	7088#	7097#	7099#
7104#	7112#	7114#	7121#	7123#	7127#	7138#	7140#	7149#	7151#	7155#	7164#	7172#	7174#	7178#
7189#	7191#	7197#	7206#	7211#	7223#	7225#	7230#	7238#	7240#	7247#	7249#	7253#	7265#	7267#
7272#	7280#	7282#	7289#	7291#	7295#	7300#	7311#	7313#	7318#	7326#	7328#	7335#	7337#	7341#
7346#	7357#	7359#	7364#	7372#	7374#	7381#	7383#	7390#	7392#	7399#	7401#	7405#	7412#	7419#
7423#	7425#	7432#	7434#	7442#	7451#	7453#	7460#	7462#	7469#	7471#	7478#	7480#	7487#	7489#
7493#	7506#	7508#	7513#	7522#	7524#	7531#	7533#	7537#	7542#	7554#	7556#	7563#	7565#	7572#
7574#	7585#	7593#	7595#	7602#	7604#	7611#	7613#	7620#	7622#	7629#	7631#	7635#	7642#	7644#
7653#	7655#	7661#	7668#	7670#	7678#	7680#	7690#	7692#	7699#	7701#	7705#	7710#	7720#	7722#
7732#	7740#	7742#	7749#	7751#	7758#	7760#	7767#	7769#	7773#	7778#	7783#	7791#	7793#	7800#
7802#	7809#	7811#	7818#	7820#	7824#	7834#	7836#	7841#	7850#	7852#	7859#	7861#	7868#	7870#







CVDMAA.P11 12-DEC-80 15:59

CROSS REFERENCE TABLE -- MACRO NAMES

SETDF	1670#	2192	2232	2272	2315	2378	2563	4661	4703	8281					
SETHRD	1670#														
SETPRI	1#	1242#													
SETSF	1670#														
SETSFT	1670#														
SETVEC	1#	1242#	3552	3615	3623	3663	3771								
SLASH	1#	1242#													
STARS	1#	1242#													
SVC	1#	1240#	1241												
T\$GEN	1670#	2192	2232	2272	2315	2378	2563	4661	4703	8281					
XFER	1#	1242#													
XFERF	1#	1242#													
XFERT	1#	1242#													
\$GEDF	1670#	2586	2620	3801	3906	4011	4084	4152	4216	4229	4309	4331	4381	4434	4493
	4510	4526	4955	5000	5218	5255	5276	5308	5329	5641	5662	5685	5711	5728	5751
	5807	5823	5879	5895	6221	6257	6325	6390	6457	6522	6587	6654	6790	6848	6901
	6948	6994	7054	7087	7126	7154	7177	7196	7210	7252	7294	7340	7404	7492	7536
	7634	7660	7704	7772	7823	7873	7902	7926	7945	7959	7983	8035	8053	8084	8130
	8197	8393	8418	8443	8482	8496	8521	8546	8602	8627	8646	8673	8693	8712	8754
	8771	8796	8856	8883											
\$GEHRD	1670#														
\$GESF	1670#														
\$GESFT	1670#														
\$GTDF	1670#	2191	2231	2271	2314	2377	2562	4660	4702	8280					
\$GTHRD	1670#														
\$GTSF	1670#														
\$GTSFT	1670#														

. ABS. 040344 000

ERRORS DETECTED: 0

CVDMAA.BIN,CVDMAA.SEQ/CRF/SOL=SVC34R.MAC,CVDMAA.P11

RUN-TIME: 49 58 6 SECONDS

RUN-TIME RATIO: 165/113=1.4

CORE USED: 24K (47 PAGES)