

IDENTIFICATION

PRODUCT CODE:                   MAINDEC-11-DQKDR-A-D  
PRODUCT NAME:                   11/6X TRAPS TEST  
PROGRAM DATE:                   MARCH 1977  
MAINTAINER:                    DIAGNOSTIC GROUP  
AUTHOR:                         BRUCE BURGESS

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.

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED UNDER A            LICENSE AND MAY ONLY BE USED OR COPIED IN ACCORDANCE WITH THE        TERMS OF SUCH LICENSE.

DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL.

COPYRIGHT (C) 1977 BY DIGITAL EQUIPMENT CORPORATION

40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  
66  
67  
68  
69

TABLE OF CONTENTS

- 1. ABSTRACT
- 2. REQUIREMENTS
  - 2.1 EQUIPMENT
  - 2.2 STORAGE
  - 2.3 PRE-REQUISITE PROGRAMS
  - 2.4 EXECUTION TIME
- 3. LOADING PROCEDURE
  - 3.1 METHOD
- 4. STARTING PROCEDURE
  - 4.1 STARTING ADDRESS
  - 4.2 PROGRAM AND/OR OPERATOR ACTION
- 5. OPERATION PROCEDURE
  - 5.1 OPERATIONAL SWITCHES
  - 5.2 FUNCTION ABSTRACTS
- 6. ERRORS
  - 6.1 ERROR RECOVERY
- 7. RESTRICTIONS
- 8. PROGRAM DESCRIPTION
- 9. ACT/APT COMPATABILITY

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  
103  
104  
105  
106  
107  
108  
109  
110  
111  
112  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122  
123  
124  
125

1. ABSTRACT  
THIS IS A TEST OF ALL OPERATIONS AND INSTRUCTIONS THAT CAUSE TRAPS. ALSO TESTED ARE TRAP OVERFLOW CONDITIONS, ODDITIES OF REGISTER 6, INTERRUPTS AND THE RESET INSTRUCTIONS. ALL THE RESERVED INSTRUCTION CODES ARE TESTED OUT.
2. REQUIREMENTS
  - 2.1 EQUIPMENT  
PDP-11/6X STANDARD COMPUTER
  - 2.2 STORAGE
    - 2.2.1 PROGRAM STORAGE - THE ROUTINE USES MEMORY FROM 0000 TO 22000.
  - 2.3 PRE-REQUISITE PROGRAMS  
ALL APPLICABLE BASIC CPU PROGRAMS SHOULD BE RUN, TO VERIFY CORRECT OPERATION OF THE BASIC INSTRUCTIONS.
  - 2.4 EXECUTION TIME  
FIRST PASS (NO ITERATIONS), QUICK VERIFY=15 SECONDS
3. LOADING PROCEDURE
  - 3.1 METHOD  
PROCEDURE FOR NORMAL ABSOLUTE TAPES SHOULD BE FOLLOWED.
4. STARTING PROCEDURE
  - 4.1 STARTING ADDRESS  
THE PROGRAM STARTS AT 200.
  - 4.2 PROGRAM AND/OR OPERATOR ACTION  
LOAD PROGRAM INTO MEMORY.  
SET SWITCH REGISTER TO STARTING ADDRESS.  
LOAD ADDRESS.  
PRESS START.  
THE PROGRAM WILL IDENTIFY ITSELF AND TEST EXECUTION WILL BEGIN IT WILL PRINT AN "END OF PASS" MESSAGE AT THE END OF EACH PASS. DURING THE FIRST PASS THERE ARE NO ITERATIONS. SUBSEQUENT PASSES HAVE 15 ITERATIONS.  
IF AN ERROR IS DETECTED, THERE WILL BE A HALT.

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  
159  
160  
161  
162  
163  
164  
165  
166  
167  
168  
169  
170  
171  
172  
173  
174  
175  
176  
177  
178  
179  
180  
181

WHEN A HALT OCCURS AND IT IS NECESSARY TO SCOPE ON IT, INSERT A "JMP#R0" INSTRUCTION IN THE "LOOP" LOCATION FOLLOWING THE HALT. THE "JMP" INSTRUCTION SHOULD BRANCH YOU TO THE PREVIOUS SCOPE LOCATION.

5. OPERATION PROCEDURE

5.1 OPERATIONAL SWITCHES

NO SWITCHES ARE USED

5.2 FUNCTION ABSTRACTS

5.2.1 BEGIN SA 200

5.2.2 HLT  
---  
INDICATES THE UNIQUE ADDRESS THAT TAGS THE FAILING SUBTEST. THE INCORRECT DATA AT THE TIME OF THE FAILURE MAY OR MAY NOT BE DISPLAYED IN REGISTER ZERO, WHICH IS THE DATA REGISTER ON A HALT.

5.2.3 SCOPE  
-----  
IS A "MOV R7,R0" INSTRUCTION THAT IS PLACED BETWEEN EACH SUBTEST IN THE INSTRUCTION SECTION. IT ESTABLISHES A POINT TO WHICH THE TEST WILL LOOP BACK TO, IN CASE A SCOPE LOOP IS ENTERED.

5.2.4 LOOP  
----  
IS A "NOP" INSTRUCTION PLACED AFTER EACH "HLT". IF A SCOPE LOOP IS DESIRED REPLACE THE "LOOP" BY A "JMP #R0" INSTRUCTION (110) AND PRESS CONTINUE. COMMENTS IN THE LISTINGS ADJACENT TO "LOOP" EXPLAIN HOW TO USE THE SCOPE LOOP. THE TEST WILL LOOP BACK TO THE PREVIOUS SCOPE INSTRUCTION.

5.2.5 TRAPCATCHER  
-----  
THIS IS A SERIES OF INSTRUCTIONS DESIGNED TO DETECT AND ISOLATE UNEXPECTED TRAPS AND INTERRUPTS, THAT OCCUR IN THE TRAP AND INTERRUPT VECTOR AREA OF MEMORY.  
  
THE PRINCIPAL OF THIS ROUTINE IS: THE VECTOR ENTRANCE ADDRESS POINTS TO THE NEXT SEQUENTIAL WORD WHICH WILL CONTAIN A HALT (000000) (THIS LOCATION IS ALSO THE STATUS WORD FOR THAT VECTOR ENTRANCE. BUT THIS WILL HAVE NO EFFECT ON IT, ALSO BEING THE NEXT INSTRUCTION).  
  
IF A HALT OCCURS IN THE TRAP OR INTERRUPT VECTOR AREA, REGISTER SIX SHOULD BE EXAMINED TO DETERMINE ITS CONTENTS. THEN USE REGISTER SIX CONTENTS AS AN ADDRESS TO DETERMINE THE LOCATION THE PROGRAM WAS AT, WHEN THE INTERRUPT OR

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  
215  
216  
217  
218  
219  
220  
221  
222  
223  
224  
225  
226  
227  
228  
229  
230  
231  
232  
233  
234  
235  
236  
237

TRAP OCCURRED. (MEMORY AS SPECIFIED BY R6 CONTAINS THE PC OF THE INSTRUCTION FOLLOWING THE INSTRUCTION WHERE TRAP OCCURRED).

THE CONTENTS OF LOCATION "\$TESTN" CONTAINS THE TEST NUMBER THAT WAS BEING EXECUTED AT THE TIME OF TRAP.

6. ERRORS

6.1 ALL ERRORS WILL CAUSE A HALT.

6.2 ERROR RECOVERY

ON TRAP ERRORS - RESTART AT STARTING ADDRESS  
DEPRESS CONTINUE TO CONTINUE TEST

IF THE MACHINE GETS "HUNG" OR FAILS IN A FATAL UNPREDICTABLE MANNER, THE USER CAN FIND OUT THE TEST WHICH WAS BEING EXECUTED AT THE TIME OF FAILURE. THE CONTENTS OF LOCATION "\$TESTN" CONTAINS THE TEST NUMBER THAT WAS BEING EXECUTED AT THE TIME OF FAILURE.

USEFUL INFORMATION CAN ALSO BE GAINED BY EXAMINING THE "PC" AT WHICH THE PROGRAM HALTED AND CROSS-REFERENCING THAT "PC" IN THE PROGRAM LISTINGS.

7. RESTRICTIONS

7.1 STARTING RESTRICTION

THE PROGRAM MUST BE STARTED AT 200.

7.2 OPERATIONAL RESTRICTION

NONE

8. PROGRAM DESCRIPTION

THIS PROGRAM CHECKS THAT ON ALL TRAP OPERATIONS REGISTER 6 IS DECREMENTED THE CORRECT AMOUNT, THAT THE CORRECT PC IS SAVED ON THE STACK, THAT THE OLD CONDITION CODES AND PRIORITY ARE PLACED ON THE STACK AND THAT THE NEW STATUS AND CONDITION CODES ARE CORRECT. BOTH THE "TRAP" AND "EMT" TRAP INSTRUCTIONS ARE TESTED TO SEE THAT ALL COMBINATIONS WILL TRAP. CHECKED ALSO IS THAT ALL RESTRICTED INSTRUCTIONS WILL TRAP. VERIFICATION OF THE "TRT" INSTRUCTION (00003) WHICH IS USED FOR SOFTWARE DEBUG ROUTINES; ODT,DDT, IS DONE. ALSO, THE TRACE BIT IS CHECKED TO SEE IF IT CAUSES A TRAP. STACK OVERFLOW IS CHECKED FOR ALL TRAP INSTRUCTIONS. YELLOW AND RED ZONE VIOLATIONS ARE CHECKED. THE RTI AND RTT INSTRUCTIONS ARE CHECKED FOR CORRECT STACK



```

294      177776      CC=177776
295      000006      R00006
296      000005      P00005
297
298      076004      MED=076004      ;MAINT, EXAM & DEPOSIT
299
300
301
302      000200      .,200
303      000200      JMP      BEGIN
304      000100      .,300
305      .SHTTL      ACT11 HOOKS
306
307      ;*****
308      ;HOOKS REQUIRED BY ACT11
309      $SVPC,      .,46      ;SAVE PC
310      $ENDAD,     .,52      ;1)SET LOC,46 TO ADDRESS OF SENDAD IN ,EOP
311      $WORD,      0      ;2)SET LOC,52 TO ZERO
312      $SVP,      .,46      ; RESTORE PC
313      .SHTTL      APT MAILBOX-ETABLE
314
315      ;*****
316
317      .EVEN
318      $MAIL:      .WORD      APT MAILBOX
319      $MSGTY:     .WORD      MSGTY ;MESSAGE TYPE CODE
320      $FATAL:    .WORD      AFATAL ;FATAL ERROR NUMBER
321      $TESTM:    .WORD      ATESTM ;TEST NUMBER
322      $PASS:     .WORD      APASS ;PASS COUNT
323      $DEVCT:    .WORD      ADEVCT ;DEVICE COUNT
324      $UNIT:     .WORD      AUNIT ;I/O UNIT NUMBER
325      $MSGADR:   .WORD      AMSGADR ;MESSAGE ADDRESS
326      $MSGLEN:  .WORD      AMSGLEN ;MESSAGE LENGTH
327      $ETABLE:   .WORD      AETABLE ;APT ENVIRONMENT TABLE
328      $ENV:      .BYTE      AENV ;ENVIRONMENT BYTE
329      $ENVM:     .BYTE      AENVM ;ENVIRONMENT MODF BITS
330      $SWREG:    .WORD      ASWREG ;APT SWITCH REGISTER
331      $USWR:     .WORD      AUSWR ;USER SWITCHES
332      $CPUOP:    .WORD      ACDUOP ;CPU TYPE,OPTIONS
333      ;*
334      ;*      BITS 15-11=CPU TYPE
335      ;*      11/04=01,11/05=02,11/20=03,11/40=04,11/45=05
336      ;*      11/70=06,POD=07,Q=10
337      ;*
338      ;*      BIT 10=REAL TIME CLOCK
339      ;*      BIT 9=FLOATING POINT PROCESSOR
340      ;*      BIT 8=MEMORY MANAGEMENT
341
342      $ETEND:
343      .MEXIT
344      .SBTTL      APT PARAMETER BLOCK
345
346      ;*****
347      ;SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT
348      ;*****
349      $EX,      .,24      ;SAVE CURRENT LOCATION
350      $P,      .,44      ;SET POWER FAIL TO POINT TO START OF PROGRAM
351      $FM,     .,00      ;FOR APT START UP
  
```

```

350      000044      .,44      ;POINT TO APT INDIRECT ADDRESS PNTR.
351      000044      $APTHDR ;POINT TO APT HEADER BLOCK
352      000030      .,30      ;RESET LOCATION COUNTER
353
354      ;*****
355      ;SETUP APT PARAMETER BLOCK AS DEFINED IN THE APT-PDP11 DIAGNOSTIC
356      ;INTERFACE SPEC.
357
358      $APTHDR:   .WORD      0      ;TWO HIGH BITS OF 10 BIT MAILBOX ADDR.
359      $MADR:     .WORD      $MAIL ;ADDRESS OF APT MAILBOX (BITS 0-15)
360      $STM:      .WORD      2      ;RHH TIME OF LONGEST TEST
361      $PSTM:     .WORD      2      ;RHH TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
362      $UNIT:     .WORD      0      ;ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
363      $UNITM:    .WORD      $ETEND-$MAIL/2 ;LENGTH MAILBOX-ETABLE(WORDS)
364
365      000304      $STM=$IFSTM
366      000302      $RPDR=$FATAL
367
368
369
  
```

```

369      #01000      .L1000
370
371
372      #01000      #00000      K11      0
373      #01002      #00000      K21      0
374      #01004      #00000      K31      0
375      #01006      #00000      K41      0
376      #01010      #00000      K51      0
377      #01012      #00000      K61      0
378      #01014      #52525      K71      52525
379      #01016      #52400      K81      52400
380      #01020      #00000      K111     0
381      #01022      #00000      K121     0
382
383      #01024      #00000      NAMFLG1  0
384
385
386      #01026      #05215      #40515      #47111      MSGNAM1  .ASCIZ  <15><12>.MAINDEC=11-DOKDB+A  POP 11/6X TRAPS TEST.
387      #01034      #42584      #26583      #30461
388      #01042      #42055      #45521      #01104
389      #01050      #40455      #20040      #42120
390      #01056      #20120      #30461      #33057
391      #01064      #20130      #51124      #50101
392      #01072      #20123      #42524      #52123
393      #01104      #000
394
395      #01102      .EVEN
396
397      #01102      #12737      177777      #21540      BGIN1:  MOV      #-1,#PASSPT      ;CLEAR THE ITERATION COUNTER
398
399
400
401      #01114      #05037      #00300      BGIN1:  CLR      #MSGTY      ;MSGTY
402      #01114      #12737      #21570      #00024      MOV      #PWRDWN,24      ;SFT UP THE POWER DOWN VECTOR
403      #01122      #12737      #00340      #00026      MOV      #340,26      ;SET UP POWER DOWN PRIORITY
404      #01130      #05037      #00304      CLR      #STNM      ;STNM
405      #01134      #05037      #00302      CLR      #ERROR      ;ERROR
406      #01140      #12702      #00300      MOV      #MSGTY,R2      ;MSGTY,R2
407      #01144      #12703      #00302      MOV      #SFATAL,R3      ;SFATAL,R3
408      #01150      #12706      #21774      MOV      #RUFF,#6      ;SET UP STACK POINTER
409
410      ;PRINT MAINDEC NUMBER AND NAME ONLY THE FIRST TIME
411
412      #01154      #05737      #01024      IST      NAMFLC      ;NAME PRINTED MCE?
413      #01160      #01021      BNE      BGN3      ;IF YES DONT PRINT AGAIN
414      #01162      #05737      #01024      INC      NAMFLC      ;SET FLAG
415      #01166      #12737      #00040      #00021      HITB     #40,#ENVM      ;WILL APT ALLOW PRINTING?
416      #01174      #01013      BNE      BGN3      ;NO
417      #01176      #12700      #01026      MOV      #MSGNAM,R0      ;MSGNAM,R0
418      #01202      #05737      177564      BGN2:   TSTB     #RTPS      ;TTY READY?
419      #01206      #00375      BPL      BGN2
420      #01210      #12703      177566      MOVB    (R0),#RTPD      ;PRINT CHARACTER
421      #01214      #01372      BNE      BGN2      ;PRINT NEXT ONE IF NOT DONE
422
423      #01216      #05737      177560      BGN2A:  TSTR     #RTPS      ;WAIT FOR DONE
424      #01222      #00375      BPL      BGN2A
  
```

```

425      #01224      BGN3:
426
427      #01224      #10700      SCOPE
428
429      ;*****
430      ;TEST 1 TEST THAT A TRAP OCCURS ON A RESERVED INSTRUCTION (760P0)
431      ;*****
432      #01226      #05237      #00304      TST1:  INC      #TESTN
433      #01232      #12706      #21774      MOV      #RUFF,#6      ;STACK POINTER SETUP
434      #01236      #12737      #01264      #00010      MOV      #RFTA,#RSEVEC      ;LOAD RESERVED INST. TRAP VECTOR
435      #01240      #05037      #00012      CLR      #RSEVEC+2      ;AND STATUS
436      #01250      #00007      RESINST      ;RESERVED INSTRUCTION (SHOULD TRAP)
437
438      #01252      #05212      INC      (R2)      ;SET MESSAGE TYPE (#MSGTY) TO FATAL ERROR
439      #01254      #12713      #00001      MOV      #1,(R3) ;MOVE TO MAIL BOX (#FATAL) THE ERROR
440      #01260      #00000      HLT      ;ERROR: RESERVED INSTRUCTION FAILED TO TRAP
441      #01262      #00240      LOOP     ;REPLACE THIS INSTRUCTION BY A
442      ;"JMP #R0" (#00110) TO GET A
443      #01264      #10700      RFTA:   SCOPE
444
445      ;*****
446      ;TEST 2 TEST DECREMENT OF STACK POINTER ON A RESERVED INSTRUCTION TRAP
447      ;*****
448      #01266      #05237      #00304      TST2:  INC      #TESTN
449      #01272      #12706      #21774      MOV      #RUFF,#6      ;STACK POINTER SETUP
450      #01276      #12737      #01306      #00010      MOV      #RFTA,#RSEVEC      ;RETURN POINTER
451      #01304      #00007      RESINST      ;DO A RESERVED INSTRUCTION
452      #01306      #20427      #21770      #RFTB:  CMP      #6,#RUFF-4      ;TEST DECREMENT OF #6
453      #01312      #01405      BEQ      15
454
455      #01314      #00712      INC      (R2)      ;SET MESSAGE TYPE (#MSGTY) TO FATAL ERROR
456      #01316      #12713      #00007      MOV      #2,(R3) ;MOVE TO MAIL BOX (#FATAL) THE ERROR
457      #01322      #00000      HLT      ;STACK POINTER (#6) WAS NOT PUSHED
458      ;DOWN BY TWO WORDS WHEN A RESERVED
459      ;INSTRUCTION TRAPPED
460      #01324      #00240      LOOP     ;REPLACE THIS INSTRUCTION BY A
461      ;"JMP #R0" (#00110) TO GET A
462      #01326      #10700      16:    SCOPE
463
464      ;*****
465      ;TEST 3 TEST THAT PROPER P.C. IS SAVED ON A RESERVED INSTRUCTION TRAP
466      ;*****
467      #01330      #05237      #00304      TST3:  INC      #TESTN
468      #01334      #12706      #21774      MOV      #RUFF,#6      ;STACK POINTER SETUP
469      #01340      #12737      #01350      #00010      MOV      #RFTA,#RSEVEC      ;RETURN POINTER
470      #01346      #00007      INSTC:  RESINST      ;TRAP ON THIS INSTRUCTION
471      #01350      #22737      #01350      #21770      #RFTC:  CMP      #1,#RUFF-4      ;CHECK FOR INCREMENTED P.C.
472      #01356      #01405      BFC      15
473
474      #01360      #05212      INC      (R2)      ;SET MESSAGE TYPE (#MSGTY) TO FATAL ERROR
475      #01362      #12713      #00007      MOV      #3,(R3) ;MOVE TO MAIL BOX (#FATAL) THE ERROR
476      #01366      #00000      HLT      ;WRONG "OLD" PC WAS SAVED ON THE STACK
477      ;WHEN A RESERVED INSTR. TRAPPED
478      #01370      #00240      LOOP     ;REPLACE THIS INSTRUCTION BY A
479      ;"JMP #R0" (#00110) TO GET A
480
  
```



```

001
002 #01372 010700      18:  SCOPE                ;SCOPE LOOP AND HIT CONTINUE
003
004
005
006
007
008 #01374 005237 000304  TST4:  INC  #01TESTN
009 #01400 012706 021774      MOV  #BUFF,06
010 #01404 012737 001420 000010  MOV  #RET0,RESVEC ;SET UP
011 #01412 005037 177776      CLR  CC
012 #01416 000007      RESINST ;CLEAR CC AND PRIORITY
013 #01420 013700 021772  RETN:  MOV  #BUFF-2,00 ;TRAP ON RESERVED INSTRUCTION
014 #01424 001405      HEQ  15 ;GET SAVED STATUS & TEST FOR ALL 0'S
015 #01426 005212      INC  (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
016 #01430 012713 000004  MOV  #0,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
017 #01434 000000      HLT ;WRONG "OLD" PSM SAVED ON STACK
018 #01436 000240      LOOP ;WHEN A RESERVED INSTR. TRAPPED
019 #01440 010700      18:  SCOPE                ;EXPECT OLD PSM=0
020 #01442 012706 021774      MOV  #BUFF,06 ;REPLACE THIS INSTRUCTION BY A
021 #01446 012737 001464 000010  MOV  #RET0,RESVEC ;"JMP #0" (000110) TO GET A
022 #01454 012737 000357 177776      MOV  #357,CC ;SCOPE LOOP AND HIT CONTINUE
023 #01462 000007      RESINST ;PRE SET THE STATUS WORD
024 #01464 013700 021772  RETE:  MOV  #BUFF-2,00 ;RESERVED INSTRUCTION TRAP
025 #01470 000000      CMP  #357,00 ;GET SAVED STATUS
026 #01474 001405      BEQ  15 ;SAVED STATUS CORRECT?
027 #01476 005212      INC  (R2) ;BRANCH IF CORRECT
028 #01500 012713 000005  MOV  #0,(R3) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
029 #01504 000000      HLT ;MOVE TO MAIL BOX ($FATAL) THE ERROR
030 #01506 000240      LOOP ;WRONG "OLD" PSM SAVED ON STACK
031 #01510 010700      18:  SCOPE                ;WHEN A RESERVED INSTR. TRAPPED
032 #01512 005237 000304      TST5:  INC  #01TESTN ;EXPECT OLD PSM=357
033 #01516 012706 021774      MOV  #BUFF,06 ;REPLACE THIS INSTRUCTION BY A
034 #01522 012737 001544 000010  MOV  #RET0,RESVEC ;"JMP #0" (000110) TO GET A
035 #01530 005037 000012 177776      CLR  RESVEC+2 ;SCOPE LOOP AND HIT CONTINUE
036 #01534 012737 000357 177776      MOV  #357,CC ;PRE SET THE STATUS WORD
037 #01542 000007      RESINST ;DO A RESERVED INSTRUCTION
038 #01544 013700 177776  RETF:  MOV  CC,00 ;GET & TEST THE "NEW" STATUS WORD
039 #01550 001405      HEQ  15 ;BRANCH IF ALL 0'S
040 #01552 005212      INC  (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
041 #01554 012713 000006  MOV  #0,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
042 #01560 000000      HLT ;"NEW" PSM WAS INCORRECT WHEN A

```

```

537
538
539 #01562 000240      LOOP ;RESERVED INSTR. TRAPPED
540 #01564 005037 177776      18:  CLR  CC ;EXPECT NEW PSM=0
541 #01570 010700      SCOPE ;REPLACE THIS INSTRUCTION BY A
542 #01572 012706 021774      MOV  #BUFF,06 ;"JMP #0" (000110) TO GET A
543 #01604 012737 000357 000012  MOV  #357,RESVEC+2 ;SCOPE LOOP AND HIT CONTINUE
544 #01612 005037 177776      CLR  CC ;LOAD THE NEW STATUS WORD
545 #01616 000007      RESINST ;PRE SET THE STATUS WORD
546 #01620 013700 177776  RETG:  MOV  CC,00 ;DO A RESERVED INSTRUCTION
547 #01624 022700 000357      CMP  #357,00 ;GET THE "NEW" STATUS WORD
548 #01630 001405      HEQ  15 ;WAS "NEW" STATUS CORRECTLY LOADED
549 #01632 005212      INC  (R2) ;BRANCH IF CORRECT
550 #01634 012713 000007  MOV  #0,(R3) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
551 #01640 000000      HLT ;MOVE TO MAIL BOX ($FATAL) THE ERROR
552 #01642 000240      LOOP ;"NEW" PSM WAS INCORRECT WHEN A
553 #01644 000240      LOOP ;RESERVED INSTRUCTION TRAPPED
554 #01646 000240      LOOP ;EXPECT NEW PSM=357
555 #01648 010700      18:  SCOPE                ;REPLACE THIS INSTRUCTION BY A
556 #01652 005037 000012 000010  MOV  #RESVEC+2,RESVEC ;"JMP #0" (000110) TO GET A
557 #01656 010700      CLR  RESVEC+2 ;TO HALT AT RESVEC+2
558
559
560
561
562
563
564
565
566
567
568
569
570 #01660 005237 000304      TST6:  INC  #01TESTN
571 #01664 012706 021774      MOV  #BUFF,06 ;STACK POINTER SETUP
572 #01670 012737 001716 000034  MOV  #RET0,TRAPVEC ;LOAD TRAP VECTOR
573 #01676 005037 000036      CLR  TRAPVEC+2
574 #01682 104400      TRAP ;DO A TRAP INSTRUCTION
575 #01704 005212      INC  (R2)
576 #01706 012713 000010  MOV  #0,(R3) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
577 #01712 000000      HLT ;MOVE TO MAIL BOX ($FATAL) THE ERROR
578 #01714 000240      LOOP ;"TRAP" INSTRUCTION DID NOT TRAP
579 #01716 010700      RETAI: SCOPE ;REPLACE THIS INSTRUCTION BY A
580 #01718 010700      RETAI: SCOPE ;"JMP #0" (000110) TO GET A
581 #01720 005237 000304      TST7:  INC  #01TESTN ;SCOPE LOOP AND HIT CONTINUE
582 #01724 012706 021774      MOV  #BUFF,06 ;STACK POINTER SETUP
583 #01730 012737 001700 000034  MOV  #RET0,TRAPVEC ;RETURN POINTER
584 #01736 104400      TRAP ;DO A TRAP INSTRUCTION
585 #01740 000627 021770  RETB1: CMP  #6,IBUFF-4 ;TEST DECREMENT OF 06
586 #01744 001405      WFO  15

```

```

591
594 001746 005212      INC (R2)          ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
595 001750 012713 000011  MOV 011,(R3)     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
596 001754 000000      HLT              ;STACK POINTER (R6) DID NOT DECREMENT
597
598 001756 000244      LOOP            ;TWO WORDS WHEN "TRAP" TRAPPED
599
600
601 001760 010700      14: SCOPE      ;REPLACE THIS INSTRUCTION BY A
602
603
604
605
606 001762 005237 000304  TST10: INC 00TESTN   ;STACK POINTER SETUP
607 001766 012706 021774  MOV 0BUFF,06    ;RETURN FROM TRAP POINTER
608 001772 012737 002052 000014  MOV 0RETCL,TRAPVEC ;TRAP ON THIS INSTRUCTION
609 002000 104400      CLR CC          ;CHECK INCREMENTED P.C.
610 002002 027237 002002 021774  RETC1: CMP 0,,BUFF-4
611 002010 001405      BEQ 10
612
613 002012 005212      INC (R2)          ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
614 002014 012713 000012  MOV 012,(R3)     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
615 002020 000000      HLT              ;WRONG "OLD" PC SAVED ON THE STACK
616
617 002022 000240      LOOP            ;WHEN "TRAP" TRAPPED
618
619
620 002024 010700      14: SCOPE      ;REPLACE THIS INSTRUCTION BY A
621
622
623
624
625 002026 005237 000304  TST11: INC 00TESTN   ;DO A TRAP INSTRUCTION
626 002030 012706 021774  MOV 0BUFF,06    ;GET & TEST THE SAVED STATUS
627 002036 012737 002052 000014  MOV 0RETCL,TRAPVEC ;BRANCH IF ALL N'S
628 002044 005037 177774  CLR CC          ;CLEAR CC AND PRIORITY
629 002052 104400      TRAP           ;DO A TRAP INSTRUCTION
630 002056 013706 021772  RETD1: MOV 0BUFF-2,00 ;GET & TEST THE SAVED STATUS
631
632
633 002060 005212      INC (R2)          ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
634 002062 012713 000013  MOV 013,(R3)     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
635 002066 000000      HLT              ;WRONG "OLD" PC SAVED ON THE STACK
636
637 002070 000240      LOOP            ;WHEN "TRAP" TRAPPED, EXPECT "0"
638
639
640 002072 010700      14: SCOPE      ;REPLACE THIS INSTRUCTION BY A
641
642
643
644 002074 012706 021774  MOV 0BUFF,06    ;INITIALIZE THE STACK POINTER
645 002076 012737 002116 000034  MOV 0RETCL,TRAPVEC ;SET UP
646 002100 012737 000357 177776  MOV 0357,CC     ;PRF SET THE STATUS WORD
647 002114 104400      TRAP           ;DO A TRAP INSTRUCTION
648 002116 013706 021772  RETE1: MOV 0BUFF-2,00 ;GET THE SAVED STATUS WORD
649 002122 022700 000357  CMP 0357,00    ;WAS CORRECT STATUS SAVED ON THE STACK
650 002126 001405      BEQ 10          ;BRANCH IF CORRECT
651

```

```

649 002130 005212      INC (R2)          ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
650 002132 012713 000014  MOV 014,(R3)     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
651 002136 000000      HLT              ;WRONG "OLD" PC SAVED ON THE STACK
652
653 002140 000240      LOOP            ;WHEN "TRAP" TRAPPED, EXPECT "357"
654
655
656 002142 010700      14: SCOPE      ;REPLACE THIS INSTRUCTION BY A
657
658
659
660
661 002144 005237 000304  TST12: INC 00TESTN   ;CLEAR FUTURE PRIORITY AND CC
662 002150 012706 021774  MOV 0BUFF,06    ;PRE SET THE STATUS WORD
663 002154 012737 002176 000034  MOV 0RETCL,TRAPVEC ;TRAPVEC+2
664 002162 005037 000036  CLR TRAPVEC+2  ;PRE SET THE STATUS WORD
665 002166 012737 000357 177776  MOV 0357,CC     ;PRE SET THE STATUS WORD
666 002174 104400      TRAP           ;DO A TRAP INSTRUCTION
667 002176 013706 177776  RETF1: MOV CC,00  ;GET THE "NEW" STATUS WORD
668 002202 001405      BEQ 10          ;BRANCH IF ALL N'S
669
670
671 002204 005212      INC (R2)          ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
672 002206 012713 000015  MOV 015,(R3)     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
673 002212 000000      HLT              ;ERROR: INCORRECT "NEW" PC SAVED
674 002214 000240      LOOP            ;REPLACE THIS INSTRUCTION BY A
675
676
677 002216 005037 177776  14: CLR CC      ;"TRAP" TRAPPED, EXPECT 0
678 002222 010700      SCOPE
679
680
681 002224 012706 021774  MOV 0BUFF,06    ;LOAD "NEW" STATUS WORD
682 002230 012717 002252 000034  MOV 0RETCL,TRAPVEC ;PRE SET THE STATUS WORD
683 002236 012737 000357 000036  MOV 0357,TRAPVEC+2 ;DO A TRAP INSTRUCTION
684 002244 005037 177776  CLR CC          ;GET THE "NEW" STATUS
685 002250 104400      TRAP           ;IS IT CORRECT?
686 002256 013706 000357  RETG1: MOV CC,00  ;BRANCH IF CORRECT
687
688
689 002264 005212      INC (R2)          ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
690 002266 012713 000016  MOV 016,(R3)     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
691 002272 000000      HLT              ;INCORRECT "NEW" PC SAVED WHEN
692
693 002274 000240      LOOP            ;"TRAP" TRAPPED, EXPECT 357
694
695
696 002276 005037 177776  14: CLR CC      ;REPLACE THIS INSTRUCTION BY A
697 002302 010700      SCOPE          ;"JMP 000" (000110) TO GET A
698
699
700
701
702 002304 005237 000304  TST13: INC 00TESTN   ;SCOPE LOOP AND HIT CONTINUE
703 002310 012737 104400 002130  MOV 0TRAP,RA1   ;INITIALIZE BASE TRAP INSTRUCTION
704 002316 012737 002344 000034  MOV 0RA1,34     ;RETURN FROM TRAP TO RA1
705 002324 012706 021774  RETI1: MOV 0BUFF,06 ;SET UP STACK POINTER

```

```

705 002330 104000      PRI:  TRAP          ;TRAP INST WILL BE MODIFIED TO TRAP+377
706
707 002332 005212      INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
708 002334 012713 000037  MOV      #17,(R3)      ;MOVE TO MAIL BOX ($FATAL) THE ERROR
709 002340 000000      HLT
710
711 002342 000240      LOOP
712
713
714 002344 005237 002330  FAIL:  INC      R01          ;REPLACE THIS INSTRUCTION BY A
715 002350 022331 104777 002330  MOV      #104777,R01    ;"JMP 000" (000110) TO GET A
716 002356 103362      CMP      R01          ;SCOPE LOOP AND HIT CONTINUE
717
718
719 002360 012737 000036 000034  MOV      #36,TRAPVEC   ;INCREMENT TRAP INSTRUCTION
720 002366 005037 000036  RCL      R01          ;TRAP+377 TO UPPER LIMIT
721 002372 010700      CUI      TRAPVEC+2     ;HAVE MY TESTED ALL
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
774
775
776
777
  
```

```

722
723
724
725 002374 005237 000304      TST14:  INC      #005237    ;*****
726 002400 012706 021774      MOV      #BUFF,%6      ;TEST 14 TEST THAT A TRAP OCCURS ON AN "IOT" INSTRUCTION
727 002404 012737 002432      MOV      #RETR2,IOTVEC ;*****
728 002412 005037 000022      CLR      IOTVEC+2      ;STACK POINTER SETUP
729 002416 000000      IOT                    ;RETURN LOCATION
730
731
732 002420 005212      INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
733 002426 000000      MOV      #20,(R3)      ;MOVE TO MAIL BOX ($FATAL) THE ERROR
734 002430 000240      HLT                    ;"IOT" INSTRUCTION FAILED TO TRAP
735
736
737 002432 010700      RETA2:  SCOPE          ;REPLACE THIS INSTRUCTION BY A
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761 002434 005237 000304      TST15:  INC      #005237    ;*****
762 002440 012706 021774      MOV      #BUFF,%6      ;TEST 15 TEST DECREMENT OF STACK POINTER ON AN "IOT" TRAP
763 002444 012737 002454      MOV      #RETR2,IOTVEC ;*****
764 002452 000000      IOT                    ;STACK POINTER SETUP
765 002454 020627 021774      CMP      #0,#BUFF-4    ;RETURN POINTER
766 002460 001405      BEW     15             ;TEST DECREMENT OF %6
767
768
769 002462 005212      INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
770 002464 012713 000021      MOV      #21,(R3)      ;MOVE TO MAIL BOX ($FATAL) THE ERROR
771 002470 000000      HLT                    ;STACK POINTER DID NOT DECREMENT BY
772
773
774 002472 000240      LOOP
775
776
777 002474 010700      I6:    SCOPE          ;TWO WORDS ON AN "IOT" TRAP
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
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
  
```

T17 TEST THAT "OLD" CC AND PRIORITY ARE PLACED ON STACK, ON AN "IOT" TRAP

```
778 ;TEST 17 TEST THAT "OLD" CC AND PRIORITY ARE PLACED ON STACK, ON AN "IOT" TRAP
779 ;*****
780 #D2542 005237 000304 TST17: INC 01TESTN
781 002546 012706 021774 MOV #BUFF,06 ;SET UP
782 002552 012737 002566 000020 MOV #RET02,IOTVEC ;SET UP
783 002560 005037 177776 CLR CC ;CLEAR CC AND PRIORITY
784 002564 000004 IOT ;TRAP
785 002566 013700 021772 RET02: MOV #UFF+2,00 ;GET & TEST SAVED STATUS
786 002572 001405 BEQ 15
787
788 #A2574 005212 INC (R2) ;SET MESSAGE TYPE (#MSGTY) TO FATAL ERROR
789 002576 012713 000023 MOV #23,(R3) ;MOVE TO MAIL BOX (#FATAL) THE ERROR
790 002602 000000 HLT ;WRONG "OLD" PSM SAVED ON STACK
791 ;WHEN "IOT" TRAPPED, EXPECT B
792 002604 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
793 ;"JMP BR" (#00110) TO GET A
794 ;SCOPE LOOP AND HIT CONTINUE
795
796 #A2606 014700 18: SCOPE
797
798 #02610 012706 021774 MOV #BUFF,06 ;SET UP
799 002614 012737 002632 000020 MOV #RETE2,IOTVEC ;SET UP
800 002622 012737 000357 177776 MOV #357,CC ;PRE SET STATUS
801 002630 000004 IOT
802 002632 013700 021772 RETE2: MOV #UFF+2,00 ;GET SAVED STATUS
803 002636 022700 000357 CMP #357,00 ;SAVED STATUS CORRECT?
804 002642 001404 BEQ 15 ;BRANCH IF CORRECT
805
806 #02644 005212 INC (R2) ;SET MESSAGE TYPE (#MSGTY) TO FATAL ERROR
807 002646 012713 000024 MOV #24,(R3) ;MOVE TO MAIL BOX (#FATAL) THE ERROR
808 002652 000000 HLT ;WRONG "OLD" PSM SAVED ON STACK
809 ;WHEN "IOT" TRAPPED, EXPECT 357
810 002654 000240 18: LOOP ;REPLACE THIS INSTRUCTION BY A
811 ;"JMP BR" (#00110) TO GET A
812 ;SCOPE LOOP AND HIT CONTINUE
813
814 ;*****
815 ;TEST 20 TEST THAT "NEW" STATUS IS CORRECT ON AN "IOT" TRAP
816 ;*****
817 #02660 005237 000304 TST20: INC 01TESTN
818 002664 012706 021774 MOV #BUFF,06
819 002670 012737 002712 000020 MOV #RETF2,IOTVEC ;CLEAR FUTURE PRIORITY AND CC
820 002676 005037 000022 CLR IOTVEC+2 ;PRE SET STATUS
821 002702 012737 000357 177776 MOV #30357,CC
822 002710 000004 IOT
823 002712 013700 177776 RETF2: MOV CC,00 ;GET & TEST "NEW" STATUS
824 002716 001405 BEQ 15 ;BRANCH IF CORRECT
825
826 #02720 005212 INC (R2) ;SET MESSAGE TYPE (#MSGTY) TO FATAL ERROR
827 002722 012713 000025 MOV #25,(R3) ;MOVE TO MAIL BOX (#FATAL) THE ERROR
828 002726 000000 HLT ;WRONG "NEW" PSM LOADED WHEN
829 ;"IOT" TRAPPED, EXPECT B
830 002730 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
831 ;"JMP BR" (#00110) TO GET A
832 ;SCOPE LOOP AND HIT CONTINUE
833 002732 005037 177776 18: CLR CC
```

T20 TEST THAT "NEW" STATUS IS CORRECT ON AN "IOT" TRAP

```
834 002736 010700 SCOPE
835
836 002740 012706 021774 MOV #BUFF,06
837 002744 012737 002766 000020 MOV #RETG2,IOTVEC
838 002752 012737 000357 000022 MOV #357,IOTVEC+2 ;LOAD "NEW" STATUS
839 002760 005037 177776 CLR CC ;PRE SET STATUS
840 002764 000004 IOT
841 002766 013700 177776 RETG2: MOV CC,00 ;GET THE "NEW" STATUS
842 002772 022700 000357 CMP #357,00 ;IS "NEW" STATUS CORRECT?
843 002776 001405 BEQ 15 ;BRANCH IF CORRECT
844
845 #03000 005212 INC (R2) ;SET MESSAGE TYPE (#MSGTY) TO FATAL ERROR
846 003002 012713 000026 MOV #26,(R3) ;MOVE TO MAIL BOX (#FATAL) THE ERROR
847 003006 000000 HLT ;WRONG "NEW" PSM LOADED WHEN
848 ;"IOT" TRAPPED, EXPECT 357
849 003010 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
850 ;"JMP BR" (#00110) TO GET A
851 ;SCOPE LOOP AND HIT CONTINUE
852 003012 18:
853
854 003012 012737 000022 000020 MOV #22,IOTVEC ;RESTORE IOT TRAP VECTOR
855 003020 005037 000022 CLR IOTVEC+2 ;TO HALT AT 22
856 003024 010700 SCOPE
```

```

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
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
=====
;TEST 12 TEST THAT "OLD" CC AND PRIORITY ARE PLACED ON STACK ON A "BPT" TRAP
=====
1070 003700 005217 000304 TST32: INC 003700
1071 003704 012726 021774 MOV 003704,00
1072 003710 012737 003724 000014 MOV 003710,00
1073 003716 005037 177776 CIP 00
1074 003722 000000 BPT
1075 003728 013700 021772 HFT04: MOV 003728,00
1076 003734 001405 BEQ 18
1078 003732 005212 INC (R2)
1079 003734 012713 000042 MOV 003734,(R3)
1080 003740 000000 HLT
1082 003744 000240 LOOP
1085 003744 010700 IS: SCOPE
1087 003746 012700 021774 MOV 003746,00
1088 003752 012737 003774 000014 MOV 003752,00
1089 003760 012737 000357 177776 MOV 003760,00
1090 003766 000000 BPT
1091 003774 013700 021772 RETE1: MOV 003774,00
1092 003774 022700 000357 CMP 003774,00
1093 003780 001405 BEQ 18
1095 004002 005212 INC (R2)
1096 004004 012713 000042 MOV 004004,(R3)
1097 004010 000000 HLT
1099 004012 000240 LOOP
1102 004014 010700 IS: SCOPE
1107 004016 005217 000304 TST33: INC 004016
1108 004022 012700 021774 MOV 004022,00
1109 004026 012737 000050 000014 MOV 004026,00
1110 004034 005037 000016 000014 CIP 00
1111 004040 012717 000357 177776 MOV 004040,00
1112 004046 000000 BPT
1113 004050 013700 177776 RETF1: MOV 004050,00
1114 004054 001405 BEQ 18
1116 004056 005212 INC (R2)
1117 004060 012713 000042 MOV 004060,(R3)
1118 004064 000000 HLT
1120 004066 000240 LOOP
=====
;TEST 13 TEST THAT "NEW" STATUS IS CORRECT ON A "BPT" TRAP
=====
1123 004070 005037 177776 IS: CIP CC
1124 004074 010700 SCOPE
1126 004076 012700 021774 MOV 004076,00
1127 004100 012737 004124 000014 MOV 004100,00
1128 004110 012737 000357 000016 MOV 004110,00
1129 004116 005037 177776 CLR 00
1130 004122 000000 BPT
1131 004124 013700 177776 RETG4: MOV 004124,00
1132 004130 022700 000357 CMP 004130,00
1133 004134 001405 BEQ 18
1135 004136 005212 INC (R2)
1136 004140 012713 000042 MOV 004140,(R3)
1137 004144 000000 HLT
1139 004146 000240 LOOP
1142 004150 005037 177776 IS: CIP CC
1144 004154 012737 000016 000016 MOV 004154,00
1145 004162 005037 000016 CLR 16
1146 004166 010700 SCOPE

```

```

1123 004070 005037 177776 IS: CIP CC
1124 004074 010700 SCOPE
1126 004076 012700 021774 MOV 004076,00
1127 004100 012737 004124 000014 MOV 004100,00
1128 004110 012737 000357 000016 MOV 004110,00
1129 004116 005037 177776 CLR 00
1130 004122 000000 BPT
1131 004124 013700 177776 RETG4: MOV 004124,00
1132 004130 022700 000357 CMP 004130,00
1133 004134 001405 BEQ 18
1135 004136 005212 INC (R2)
1136 004140 012713 000042 MOV 004140,(R3)
1137 004144 000000 HLT
1139 004146 000240 LOOP
1142 004150 005037 177776 IS: CIP CC
1144 004154 012737 000016 000016 MOV 004154,00
1145 004162 005037 000016 CLR 16
1146 004166 010700 SCOPE

```

```
1140  
1149 ;*****  
1150 ;*TEST 34 TEST THAT A TRAP OCCURS ON AN ILLEGAL INSTRUCTION - JMP P0  
1151 ;*****  
1152 004170 005237 000304 TST34: INC #RTESTIN  
1153 004174 012706 021774 MOV #RUFF,06 ;STACK POINTER SETUP  
1154 004200 012737 004226 000004 MOV #RETAS,ERRVEC ;RETURN LOCATION  
1155 004206 005037 000006 CLR ERRVEC+2  
Z 1156 004212 000100 JMP 00 ;ILLEGAL INSTRUCTION, SHOULD TRAP  
1157  
1158 004214 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
1159 004216 012711 000046 MOV #46,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
1160 004222 000000 HLT ;ERROR: SAVED PSW IS INCORRECT  
1161 004224 004214 LOOP ;ON ILLEGAL INSTR. TRAP, EXPECT 0  
1162 ;REPLACE THIS INSTRUCTION BY A  
1163 ;"JMP 000" (000110) TO GET A  
1164 004226 010700 RETAS: SCOPE ;SCOPE LOOP AND HIT CONTINUE  
1165  
1166 ;*****  
1167 ;*TEST 35 TEST DECREMENT OF STACK POINTER ON AN ILLEGAL INSTRUCTION TRAP  
1168 ;*****  
1169 004230 005237 000304 TST35: INC #RTESTIN  
1170 004234 012706 021774 MOV #RUFF,06 ;STACK POINTER SETUP  
1171 004240 012737 004250 000004 MOV #RETS5,ERRVEC ;RETURN POINTER  
Z 1172 004246 000100 JMP 00 ;RESERVED INSTRUCTION  
1173 004250 020027 021770 RETB5: CMP #0,RUFF-4 ;TEST DECREMENT OF R6  
1174 004254 001405 BEQ 10  
1175  
1176 004256 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
1177 004260 012711 000047 MOV #47,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
1178 004264 000000 HLT ;STACK POINTER DID NOT DECREMENT  
1179 ;BY TWO WORDS ON ILLEGAL INSTR. TRAP  
1180 004266 000240 LOOP ;REPLACE THIS INSTRUCTION BY A  
1181 ;"JMP 000" (000110) TO GET A  
1182 ;SCOPE LOOP AND HIT CONTINUE  
1183 004270 010700 10: SCOPE  
1184  
1185 ;*****  
1186 ;*TEST 36 TEST THAT PROPER P.C. IS SAVED ON AN ILLEGAL INSTRUCTION TRAP  
1187 ;*****  
1188 004272 005237 000304 TST36: INC #RTESTIN  
1189 004276 012706 021774 MOV #RUFF,06 ;STACK POINTER SETUP  
1190 004302 012737 004312 000004 MOV #RETC5,ERRVEC ;RETURN FROM TRAP POINTER  
Z 1191 004310 000100 JMP 00 ;TRAP ON THIS INSTRUCTION  
1192 004312 022737 004312 021770 RETC5: CMP #0,RUFF+4 ;CHECK FOR INCREMENTED P.C.  
1193 004320 001405 BEQ 10  
1194  
1195 004322 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
1196 004324 012711 000054 MOV #50,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
1197 004330 000000 HLT ;ERROR: "OLD" PC SAVED ON STACK  
1198 ;ON ILLEGAL INSTR. TRAP  
1199 004332 000240 LOOP ;REPLACE THIS INSTRUCTION BY A  
1200 ;"JMP 000" (000110) TO GET A  
1201 ;SCOPE LOOP AND HIT CONTINUE  
1202 004334 010700 10: SCOPE  
1203
```

```
1204 ;*****  
1205 ;*TEST 37 TEST THAT "OLD" CC AND PRIORITY ARE SAVED ON AN ILLEGAL INSTR. TRAP  
1206 ;*****  
1207 004336 005237 000304 TST37: INC #RTESTIN  
1208 004342 012706 021774 MOV #RUFF,06 ;SET UP  
1209 004346 012737 004362 000004 MOV #RETD5,ERRVEC ;SET UP  
Z 1210 004354 005037 177776 CLR CC ;CLEAR CC AND PRIORITY  
1211 004360 000100 JMP 00 ;TRAP  
1212 004362 013700 021772 RETD5: MOV RUFF-2,00 ;GET THE SAVED STATUS  
1213 004366 001405 BEQ 10 ;IS IT CORRECT  
1214 ;BRANCH IF SAVED STATUS IS CORRECT  
1215  
1216 004370 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
1217 004372 012711 000051 MOV #51,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
1218 004376 000000 HLT ;ERROR: SAVED PSW IS INCORRECT  
1219 ;ON ILLEGAL INSTR. TRAP, EXPECT 0  
1220 004400 000240 LOOP ;REPLACE THIS INSTRUCTION BY A  
1221 ;"JMP 000" (000110) TO GET A  
1222 ;SCOPE LOOP AND HIT CONTINUE  
1223 004402 010700 10: SCOPE  
1224  
1225 004404 012706 021774 MOV #RUFF,06 ;SET UP  
1226 004410 012737 004426 000004 MOV #RETE5,ERRVEC ;SET UP  
Z 1227 004424 000100 JMP 00 ;PRE SET STATUS  
1228 004426 013700 021772 RETE5: MOV RUFF-2,00 ;GET THE SAVED STATUS  
1229 004432 022700 000357 CMP #357,00 ;IS IT CORRECT  
1230 004436 001405 BEQ 10 ;BRANCH IF SAVED STATUS IS CORRECT  
1231  
1232 004440 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
1233 004442 012711 000052 MOV #52,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
1234 004446 000000 HLT ;ERROR: SAVED PSW IS INCORRECT  
1235 ;ON ILLEGAL INSTR. TRAP, EXPECT 0  
1236 004450 000240 LOOP ;REPLACE THIS INSTRUCTION BY A  
1237 ;"JMP 000" (000110) TO GET A  
1238 ;SCOPE LOOP AND HIT CONTINUE  
1239 004452 010700 10: SCOPE  
1240  
1241 ;*****  
1242 ;*TEST 40 TEST THAT "NEW" STATUS IS CORRECT ON AN ILLEGAL INSTR. TRAP  
1243 ;*****  
1244 004454 005237 000304 TST40: INC #RTESTIN  
1245 004460 012706 021774 MOV #RUFF,06 ;SET UP  
1246 004464 012737 004506 000004 MOV #RETF5,ERRVEC ;SET UP  
Z 1247 004472 005037 000006 CLR ERRVEC+2 ;CLEAR FUTURE PRIORITY AND CC  
1248 004476 012737 000357 177776 MOV #357,CC ;PRE SET STATUS  
1249 004504 000100 JMP 00  
1250 004506 013700 177776 RETF5: MOV CC,00 ;GET & TEST "NEW" STATUS  
1251 004512 001405 BEQ 10  
1252  
1253 004514 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
1254 004516 012711 000053 MOV #53,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
1255 004522 000000 HLT ;ERROR: "NEW" PSW IS INCORRECT  
1256 ;ON ILLEGAL INSTR. TRAP, EXPECT 0  
1257 004524 000240 LOOP ;REPLACE THIS INSTRUCTION BY A  
1258 ;"JMP 000" (000110) TO GET A  
1259 ;SCOPE LOOP AND HIT CONTINUE  
1260
```

```

1260 004526 005037 177776 10: CLR CC
1261 004532 010700 SCOPE
1262
1263 004534 012706 021774 MOV #BUFF,06
1264 004540 012737 004562 000004 MOV #RETG5,ERRVEC ;STACK POINTER SETUP
1265 004546 012737 000357 000006 MOV #357,ERRVEC+2 ;RETURN LOCATION
1266 004554 005037 177776 CLR CC ;LOAD "NEW" STATUS
Z 1267 004560 000100 JMP 00 ;PRE SET STATUS
1268 004562 013300 177776 RETG5: MOV CC,00 ;GET THE "NEW" STATUS
1269 004566 022700 000357 CMP #357,00 ;IS IT CORRECT
1270 004572 001005 BFC 10 ;BRANCH IF CORRECT
1271
1272 004574 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1273 004576 012713 000054 MOV #54,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1274 004602 000000 HLT ;ERROR: "NEW" STATUS IS INCORRECT
1275 ;ON ILLEGAL INSTR. TRAP, EXPECT 357
1276 004604 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
1277 ;"JMP 00" (000110) TO GET A
1278 ;"JMP 00" (000110) TO GET A
1279 004606 005037 177776 10: CLR CC ;SCOPE LOOP AND HIT CONTINUE
1280 004612 010700 SCOPE
  
```

```

1281
1282
1283 ;*****
1284 ;*TEST 41 TEST THAT A TRAP OCCURS ON ALL ILLEGAL INSTRUCTION - JSR R0,R0
1285 ;*****
1286 004614 005237 000304 TST41: INC #0TESTN
1287 004620 012706 021774 MOV #BUFF,06 ;STACK POINTER SETUP
1288 004624 012737 004652 000004 MOV #RET5,ERRVEC ;RETURN LOCATION
1289 004632 005037 000006 CLR ERRVEC+2
Z 1290 004636 000000 JSR 00,00 ;ILLEGAL INSTRUCTION (SHOULD TRAP)
1291
1292 004640 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1293 004642 012713 000055 MOV #55,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1294 004646 000000 HLT ;ILLEGAL INSTRUCTION FAILED TO TRAP
1295 004650 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
1296 ;"JMP 00" (000110) TO GET A
1297 004652 010700 RETN5: SCOPE ;SCOPE LOOP AND HIT CONTINUE
1298
1299
1300 ;TEST DECREMENT OF STACK POINTER ON A TRAP OPERATION
1301 004654 012706 021774 MOV #BUFF,06 ;STACK POINTER SETUP
Z 1302 004660 012737 004670 000004 MOV #RETJ,ERRVEC ;RETURN POINTER
1303 004666 004000 JSR 00,00 ;RESERVED INSTRUCTION
1304 004670 020627 021774 RETJ: CMP #0,#BUFF-4 ;TEST DECREMENT OF 4b
1305 004674 001005 BFC 10
1306
1307 004676 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1308 004700 012713 000056 MOV #56,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1309 004704 000000 HLT ;R0 DID NOT DECREMENT BY 2 WORDS
1310 004706 000240 LOOP ;ON ILLEGAL INSTR. TRAP
1311 ;REPLACE THIS INSTRUCTION BY A
1312 ;"JMP 00" (000110) TO GET A
1313 004710 010700 10: SCOPE ;SCOPE LOOP AND HIT CONTINUE
1314
1315
1316 ;*****
1317 ;*TEST 42 TEST THAT PROPER P.C. IS SAVED ON AN ILLEGAL INSTR. TRAP
1318 ;*****
1319 004712 005237 000304 TST42: INC #0TESTN
1320 004716 012706 021774 MOV #BUFF,06 ;STACK POINTER SETUP
1321 004722 012737 004732 000004 MOV #RETJ,ERRVEC ;RETURN FROM TRAP POINTER
Z 1322 004732 022737 004732 021774 INSTR: JSP 00,00 ;TRAP ON THIS INSTRUCTION
1323 004740 001005 RETK: CMP #INSTR+2,BUFF-4 ;CHECK FOR INCREMENTED P.C.
1324 004740 001005 BFC 10
1325
1326 004742 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1327 004744 012713 000057 MOV #57,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1328 004750 000000 HLT ;WRONG "OLD" PC SAVED ON STACK
1329 004752 000240 LOOP ;ON ILLEGAL INSTR. TRAP
1330 ;REPLACE THIS INSTRUCTION BY A
1331 ;"JMP 00" (000110) TO GET A
1332 004754 010700 10: SCOPE ;SCOPE LOOP AND HIT CONTINUE
1333
1334
1335 ;*****
1336 ;*TEST 43 TEST THAT "OLD" CC AND PRIORITY ARE SAVED ON AN ILLEGAL INSTR. TRAP
  
```

```

1337 004756 005217 000304 TST43: INC #RSTRTN
1338 004762 012706 021774 MOV #RUFF,06 ;SET UP
1339 004766 012737 005002 000004 MOV #RETR,ERRVEC ;SET UP
1340 004774 005037 177776 CLR CC ;CLEAR CC AND PRIORITY
Z 1341 005000 004000 JSR 00,00
1342 005002 013700 021777 RETN: MOV #RUFF-2,00 ;GET & TEST SAVED STATUS
1343 005006 001405 BEQ 16 ;BRANCH IF ALL 0'S
1344
1345 005010 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
1346 005012 012713 000004 MOV #00,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
1347 005016 000000 HLT ;ERROR: INCORRECT PSM SAVED ON STACK
1348 ;ON ILLEGAL INSTR. TRAP, EXPECT #
1349 005020 000704 LOOP ;REPLACE THIS INSTRUCTION BY A
1350 ;"JMP #R0" (000110) TO GET A
1351 ;"JMP #R0" (000110) TO GET A
1352 005022 010704 16: SCOPE ;SCOPE LOOP AND HIT CONTINUE
1353
1354 005024 012706 021774 MOV #RUFF,06 ;SET UP
1355 005030 012737 005006 000004 MOV #RETR,ERRVEC ;SET UP
1356 005036 012737 000357 177776 MOV #R37,CC ;PRE SET STATUS
Z 1357 005044 000000 CLR CC
1358 005046 013700 021777 RETN: MOV #RUFF-2,00 ;GET SAVED STATUS
1359 005052 022700 000357 CMP #R37,00 ;IS IT CORRECT
1360 005056 001405 BTL 16
1361
1362 005060 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
1363 005062 012713 000004 MOV #01,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
1364 005066 000000 HLT ;ERROR: INCORRECT PSM SAVED ON STACK
1365 ;ON ILLEGAL INSTR. TRAP, EXPECT 357
1366 005070 000210 LOOP ;REPLACE THIS INSTRUCTION BY A
1367 ;"JMP #R0" (000110) TO GET A
1368 ;"JMP #R0" (000110) TO GET A
1369 005072 010704 16: SCOPE ;SCOPE LOOP AND HIT CONTINUE
1370
1371 ;*****
1372 ;=TEST 44 TEST THAT "NEW" STATUS IS CORRECT ON AN ILLEGAL INSTR. TRAP
1373 ;*****
1374 005074 005237 000304 TST44: INC #RSTRTN
1375 005100 012706 021774 MOV #RUFF,06
1376 005104 012737 005126 000004 MOV #RETR,ERRVEC
1377 005112 005037 000006 CLR #ERRVEC+2 ;CLEAR FUTURE PRIORITY AND CC
Z 1378 005116 012737 000357 177776 MOV #R37,CC ;PRE SET STATUS
1379 005124 004000 JSR 00,00
1380 005126 013700 021777 RETN: MOV #RUFF-2,00 ;GET & TEST "NEW" STATUS
1381 005132 001405 BEQ 16 ;BRANCH IF ALL 0'S
1382
1383 005134 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
1384 005136 012713 000004 MOV #02,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
1385 005142 000000 HLT ;ERROR: "NEW" PSM IS INCORRECT
1386 ;ON ILLEGAL INSTR. TRAP
1387 005144 000210 LOOP ;REPLACE THIS INSTRUCTION BY A
1388 ;"JMP #R0" (000110) TO GET A
1389 ;"JMP #R0" (000110) TO GET A
1390 005146 005037 177776 16: CLR CC
1391 005152 010704 16: SCOPE
1392

```

```

1393 005154 012706 021774 MOV #RUFF,06
1394 005160 012737 005202 000004 MOV #RETR,ERRVEC
1395 005166 012737 000357 000006 MOV #R37,ERRVEC+2 ;LOAD "NEW" STATUS
Z 1396 005174 005037 177776 CLR CC ;PRE SET STATUS
1397 005200 004000 JSR 00,00
1398 005202 013700 021777 RETN: MOV #RUFF,06 ;GET THE "NEW" STATUS
1399 005206 022700 000357 CMP #R37,00 ;IS IT CORRECT
1400 005212 001405 BFO 16 ;BRANCH IF CORRECT
1401
1402 005214 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
1403 005216 012713 000004 MOV #03,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
1404 005222 000000 HLT ;ERROR: "NEW" PSM IS INCORRECT
1405 ;ON ILLEGAL INSTR. TRAP
1406 005224 000210 LOOP ;REPLACE THIS INSTRUCTION BY A
1407 ;"JMP #R0" (000110) TO GET A
1408 ;"JMP #R0" (000110) TO GET A
1409 005226 005037 177776 16: CLR CC
1410 005232 010704 16: SCOPE
1411
1412 ;*****
1413 ;=TEST 45 TEST THAT A TRAP OCCURS ON AN ILLEGAL ADDRESS (ODD)
1414 ;*****
1415
1416 005234 005237 000304 TST45: INC #RSTRTN
1417 005240 012706 021774 MOV #RUFF,06 ;STACK POINTER SETUP
1418 005244 012737 005274 000004 MOV #RETR,ERRVEC ;RETURN LOCATION
1419 005252 005037 000006 CLP #ERRVEC+7
1420 005256 005137 000001 TST 1 ;ILL.ADDRS. (ODD ADDRESS ON WORD INST.)
1421
1422 005262 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
1423 005264 012713 000004 MOV #04,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
1424 005270 000000 HLT ;ILLEGAL ADDRESS DID NOT TRAP
1425 005272 000210 LOOP ;REPLACE THIS INSTRUCTION BY A
1426 ;"JMP #R0" (000110) TO GET A
1427 ;"JMP #R0" (000110) TO GET A
1428 005274 010704 RETN: SCOPE ;SCOPE LOOP AND HIT CONTINUE
1429

```



```

1432
1431
1432
1433
1434 005276 005237 000304 TST46: INC 005TESTN
1435 005382 012706 021774 MOV 0BUFF,06 ;STACK POINTER SETUP
1436 005306 012737 005328 000004 MOV 0RETU,ERRVEC ;RETURN POINTER
1437 005314 005737 000001 TST 1 ;RESERVED INSTRUCTION
1438 005320 020627 021770 RETO: CMP 06,0BUFF-4 ;TEST DECREMENT OF 06
1439 005324 001405 BEQ 15
1440
1441 005326 005232 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
1442 005330 012713 000005 MOV 005,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
1443 005334 000000 HLT ;R0 DID NOT DECREMENT BY 2 WORDS
1444
1445 005336 000714 LOOP ;ON AN ODD ADDRESS TRAP
1446 ;REPLACE THIS INSTRUCTION BY A
1447 ;"JMP 000" (000110) TO GET A
1448 005340 010700 15: SCOPE ;SCOPE LOOP AND HIT CONTINUE
1449
1450
1451
1452
1453 005342 005237 000304 TST47: INC 005TESTN
1454 005346 012706 021774 MOV 0BUFF,06 ;STACK POINTER SETUP
1455 005352 012737 005364 000004 MOV 0RETU,ERRVEC ;RETURN FROM TRAP POINTER
1456 005360 005737 000001 TST 1 ;TRAP ON THIS INSTRUCTION
1457 005364 022737 005364 021770 RETR: CMP 0,,0UFF-4 ;CHECK FOR INCREMENTED P.C.
1458 005372 001405 BEQ 15
1459
1460 005374 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
1461 005376 012713 000006 MOV 006,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
1462 005402 000000 HLT ;P000C "OLD" PC SAVED ON STACK
1463 ;ON ODD ADDRESS TRAP
1464 005404 000714 LOOP ;REPLACE THIS INSTRUCTION BY A
1465 ;"JMP 000" (000110) TO GET A
1466 ;SCOPE LOOP AND HIT CONTINUE
1467 005406 010700 15: SCOPE
1468
1469
1470
1471
1472 005414 005237 000304 TST50: INC 005TESTN
1473 005414 012706 021774 MOV 0BUFF,06 ;SET UP
1474 005420 012737 005436 000004 MOV 0RETS,ERRVEC ;SET UP
1475 005426 005037 177776 CLR CC ;CLEAR CC AND PRIORITY
1476 005432 005737 000001 TST 1
1477 005436 013700 021772 RETS: MOV 0BUFF-2,00 ;GET & TEST SAVED STATUS ON STACK
1478 005442 001405 BEQ 15 ;BRANCH IF SAVED STATUS IS CORRECT
1479
1480 005444 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
1481 005446 012713 000007 MOV 007,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
1482 005452 000000 HLT ;ERROR! SAVED PSW IS INCORRECT
1483 005454 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
1484 ;"JMP 000" (000110) TO GET A
1485 ;SCOPE LOOP AND HIT CONTINUE

```

```

1486 005456 010700 15: SCOPE
1487
1488 005460 012706 021774 MOV 0BUFF,06 ;SET UP
1489 005464 012737 005504 000004 MOV 0RETS,ERRVEC ;SET UP
1490 005472 012737 000357 177776 MOV 0357,CC ;PRE SET STATUS
1491 005500 005737 000001 TST 1
1492 005504 013700 021772 RETR: MOV 0BUFF-2,00 ;GET THE SAVED STATUS OFF STACK
1493 005510 022700 000357 CMP 0357,00 ;IS IT CORRECT
1494 005514 001405 BEQ 15 ;BRANCH IF CORRECT
1495
1496 005516 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
1497 005520 012713 000070 MOV 070,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
1498 005524 000000 HLT ;ERROR! SAVED PSW ON STACK IS INCORRECT
1499 005526 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
1500 ;"JMP 000" (000110) TO GET A
1501 ;SCOPE LOOP AND HIT CONTINUE
1502 005530 010700 15: SCOPE
1503
1504
1505
1506
1507 005532 005237 000304 TST51: INC 005TESTN
1508 005536 012706 021774 MOV 0BUFF,06 ;LOAD "NEW" STATUS
1509 005542 012737 005566 000004 MOV 0RETU,ERRVEC ;LOAD "NEW" STATUS
1510 005550 005037 000006 CLR ERRVEC+2 ;CLEAR FUTURE PRIORITY AND CC
1511 005554 012737 000357 177776 MOV 0357,CC ;PRE SET STATUS
1512 005562 005737 000001 TST 1
1513 005566 013700 177776 RETU: MOV CC,00 ;GET & TEST THE "NEW" STATUS
1514 005572 001405 BEQ 15 ;BRANCH IF CORRECT
1515
1516 005574 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
1517 005576 012713 000071 MOV 071,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
1518 005602 000000 HLT ;ERROR! "NEW" PSW IS INCORRECT
1519 005604 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
1520 ;"JMP 000" (000110) TO GET A
1521 ;SCOPE LOOP AND HIT CONTINUE
1522 005606 005037 177776 15: CLR CC
1523 005612 010700 15: SCOPE
1524
1525 005614 012706 021774 MOV 0BUFF,06
1526 005620 012737 005640 000004 MOV 0RETU,ERRVEC
1527 005626 012737 000357 000006 MOV 0357,ERRVEC+2 ;LOAD "NEW" STATUS
1528 005634 005737 000001 TST 1
1529 005640 013700 177776 RETV: MOV CC,00 ;GET THE "NEW" STATUS
1530 005644 022700 000357 CMP 0357,00 ;IS IT CORRECT
1531 005650 001405 BEQ 15 ;BRANCH IF CORRECT
1532
1533 005652 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
1534 005654 012713 000072 MOV 072,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
1535 005660 000000 HLT ;ERROR! "NEW" STATUS IS INCORRECT
1536 005662 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
1537 ;"JMP 000" (000110) TO GET A
1538 ;SCOPE LOOP AND HIT CONTINUE
1539 005664 005037 177776 15: CLR CC
1540 005670 010700 15: SCOPE
1541

```

T52 TEST THAT AN ODDSOURCE INTERMEDIATE ADDRESS CAUSES AN ODD ADDRESS TRAP

```
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
T52:
INC R18TESTN
MOV RUFF,86 ;SET UP STACK POINTER
MOV RDAER4,4 ;LOAD ERROR VECTOR
CLR 6
MOV R1,81 ;LOAD INDX REGISTER
ADD R(1),80 ;SRC ADPS (R1) IS ODD
INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
MOV R73,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
HLT ;ERROR: ODD ADRES ERROR FAILED TO TRAP
LOOP ;REPLACE THIS INSTRUCTION BY A
;"JMP R00" (000110) TO GET A
;SCOPE LOOP AND HIT CONTINUE
DAER0: SCOPE
MOV RUFF,86
MOV RDAER4,4
MOV RUFF-1,81
BTCR R(1),80 ;SRC INT ADRS (R1) IS ODD
INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
MOV R74,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
HLT ;ERROR: ODD ADPS IN DST FAILED TO TRAP
LOOP ;REPLACE THIS INSTRUCTION BY A
;"JMP R00" (000110) TO GET A
;SCOPE LOOP AND HIT CONTINUE
DAER1: SCOPE
T53:
INC R18TESTN
MOV RUFF,86
MOV RDAER4,4
MOV R1,100
CLR 81
MOV R100(1),80 ;SRC FINAL ADRS IS ODD
INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
MOV R75,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
HLT ;ERROR: ODD FINAL SRC ADRS FAILED TO TRAP
LOOP ;REPLACE THIS INSTRUCTION BY A
;"JMP R00" (000110) TO GET A
;SCOPE LOOP AND HIT CONTINUE
DAER2: SCOPE
T54:
INC R18TESTN
MOV RUFF,86
MOV RDAER4,4
MOV R1,81
T54 TEST THAT AN ODD DEST INTERMEDIATE ADDRESS CAUSES AN ODD ADRS TRAP
```

T54 TEST THAT AN ODD DEST INTERMEDIATE ADDRESS CAUSES AN ODD ADRS TRAP

```
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
T54:
XOR R,0(1) ;DST INT ADPS (R1) IS ODD
INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
MOV R76,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
HLT ;ERROR: ODD ADRS ERROR FAILED TO TRAP
LOOP ;REPLACE THIS INSTRUCTION BY A
;"JMP R00" (000110) TO GET A
;SCOPE LOOP AND HIT CONTINUE
DAER3: SCOPE
MOV RUFF,86
MOV RDAER4,4
CLR 6
CMPB (1)+,0(1)+ ;DST INT ADRS IS ODD ((R1) =1)
INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
MOV R77,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
HLT ;ERROR: ODD ADRS ERROR FAILED TO TRAP
LOOP ;REPLACE THIS INSTRUCTION BY A
;"JMP R00" (000110) TO GET A
;SCOPE LOOP AND HIT CONTINUE
DAER4: SCOPE
T55:
INC R18TESTN
MOV RUFF,86
MOV RDAER4,4
CLR 81
SXT R100(1) ;DST FINAL ADPS IS ODD
INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
MOV R100,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
HLT ;ERROR: DST ODD ADRS ERR FAILED TO TRAP
LOOP ;REPLACE THIS INSTRUCTION BY A
;"JMP R00" (000110) TO GET A
;SCOPE LOOP AND HIT CONTINUE
DAER5: SCOPE
T56:
INC R18TESTN
MOV RUFF,86 ;INITIALIZE THE STACK POINTER
MOV RDAER4,4
CLR 81
MOV RTEMP1,8
MOV RTEMP2,2
MOV R123000,TEMP
MOV R177246,TEMP+2
CMPB R(1)+,R(1)+ ;COMP. LOC. (MYFF) TEMP+1 & TEMP+2
BEQ 15
INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
T56 TEST THAT AN ODD DEST FINAL ADRS WILL CAUSE AN ODD ADRS TRAP
```

```
1654 006264 012713 000101      MOV      #101,(R1)      ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1655 006270 000000      HLT                      ;COMPARE INSTRUCTION FAILED
1656 006272 000240      LOOP                   ;REPLACE THIS INSTRUCTION BY A
1657                                     ;"JMP BR#" (000110) TO GET A
1658                                     ;SCOPE LOOP AND HIT CONTINUE
1659 006274 022701 000004      18:    CMP      #4,#1      ;DID REGISTER INCREMENT PROPERLY
1660 006300 001405      BEQ                      ;
1661                                     ;
1662 006302 005212      INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1663 006304 012713 000102      MOV      #102,(R1)     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1664 006310 000000      HLT                      ;ERROR! PI DID NOT INC. BY +2 TWICE
1665 006312 000240      LOOP                   ;REPLACE THIS INSTRUCTION BY A
1666                                     ;"JMP BR#" (000110) TO GET A
1667                                     ;SCOPE LOOP AND HIT CONTINUE
1668 006314 000405      28:    BR      0AEP0
1669                                     ;
1670 006316 005212      INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1671 006320 012713 000103      MOV      #103,(R1)     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1672 006324 000000      HLT                      ;ERROR! ODD ADDRESS ERROR
1673 006326 000240      LOOP                   ;REPLACE THIS INSTRUCTION BY A
1674                                     ;"JMP BR#" (000110) TO GET A
1675                                     ;SCOPE LOOP AND HIT CONTINUE
1676 006330 010700      0AER0: SCOPE
1677                                     ;
1678                                     ;*****
1679                                     ;*TEST 57 TEST THAT SWAB ODD ADDRESS CAUSES AN ODD ADDRESS TRAP
1680                                     ;*****
1681 006332 005237 000304      TST57: INC      #0TESTN
1682 006336 012706 021774      MOV      #RUFF,#6
1683 006342 012737 000366 000004      MOV      #0AER7,4
1684 006350 000337 021415      SWAB     TEMP+1        ;DO SWAB USING AN ODD ADDRESS
1685                                     ;
1686 006354 005212      INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1687 006356 012713 000104      MOV      #104,(R1)     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1688 006362 000000      HLT                      ;ERROR! FAILED TO TRAP
1689 006364 000240      LOOP                   ;REPLACE THIS INSTRUCTION BY A
1690                                     ;"JMP BR#" (000110) TO GET A
1691                                     ;SCOPE LOOP AND HIT CONTINUE
1692 006366      0AER7:
1693                                     ;
1694 006366 012737 000006 000004      MOV      #ERRVEC+7,#ERRVEC;RESTORE ODD ADDRESS ERROR TO
1695 006374 005037 000006      CLR      ERRVEC+2      ;HALT AT ERRVEC+2
1696 006400 010700      SCOPE
1697
```

```
1698
1699
1700                                     ;*****
1701                                     ;*TEST 60 TEST THAT SETTING THE "T" BIT WILL CAUSE A TRAP TO 14
1702                                     ;*****
1703 006402 005237 000304      TST60: INC      #0TESTN
1704 006406 012706 021774      MOV      #RUFF,#6
1705 006412 005001      CLR      #1
1706 006414 012737 000454 000014      MOV      #RETAT,TRTVEC ;SET UP TO TRAP TO 14
1707 006422 005037 000016      CLN      TRTVEC+2
1708 006426 012746 000020      MOV      #20,-(6)      ;PUSH "T" BIT ON THE STACK
1709 006432 012746 000440      MOV      #,-(6)        ;PUSH PC ON THE STACK
1710 006436 000002      RTI                      ;SET "T" BIT
1711 006440 005201      INC      R1              ;TRAP HERE
1712                                     ;
1713 006442 005212      INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1714 006444 012713 000105      MOV      #105,(R1)     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1715 006450 000000      HLT                      ;TRACE RTI DOES NOT TRAP
1716 006452 000240      LOOP                   ;REPLACE THIS INSTRUCTION BY A
1717                                     ;"JMP BR#" (000110) TO GET A
1718                                     ;SCOPE LOOP AND HIT CONTINUE
1719 006454 005701      RETAT: TST      R1
1720 006456 001405      BEQ                      ;DID T BIT TRAP RIGHT? AFTER RTI?
1721                                     ;YFS
1722 006460 005212      INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1723 006462 012713 000106      MOV      #106,(R1)     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1724 006466 000000      HLT                      ;T BIT TRAP DID NOT OCCUR
1725                                     ;RIGHT AFTER RTI (ONE INSTR
1726                                     ;WAS ALLOWED AFTER RTI)
1727 006470 000240      LOOP                   ;REPLACE THIS INSTRUCTION BY A
1728                                     ;"JMP BR#" (000110) TO GET A
1729                                     ;SCOPE LOOP AND HIT CONTINUE
1730 006472 010700      18:    SCOPE
1731                                     ;
1732                                     ;*****
1733                                     ;*TEST 61 TEST STACK POINTER DECREMENTS ON A T BIT TRAP
1734                                     ;*****
1735 006474 005237 000304      TST61: INC      #0TESTN
1736 006500 012706 021774      MOV      #RUFF,#6
1737 006504 012737 000524 000014      MOV      #RETBT,TRTVEC ;PUSH "T" BIT ON THE STACK
1738 006512 012746 000020      MOV      #20,-(6)      ;PUSH PC ON THE STACK
1739 006516 012746 000524      MOV      #,-(6)
1740 006522 000002      RTI                      ;SET "T" BIT
1741 006524 020627 021770      RETBT: CMP      #6,#BUFF-4
1742 006530 001405      BEQ                      ;
1743                                     ;
1744 006532 005212      INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1745 006534 012713 000107      MOV      #107,(R1)     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1746 006540 000000      HLT                      ;STACK POINTER WAS NOT PUSHED TWICE BY T BIT TRAP
1747 006542 000240      LOOP                   ;REPLACE THIS INSTRUCTION BY A
1748                                     ;"JMP BR#" (000110) TO GET A
1749                                     ;SCOPE LOOP AND HIT CONTINUE
1750 006544 010700      18:    SCOPE
1751                                     ;
1752                                     ;*****
1753                                     ;*TEST 62 TEST FOR PROPER PC ON STACK ON A T BIT TRAP
1754                                     ;*****
1755 006546 005237 000304      TST62: INC      #0TESTN
1756 006552 012706 021774      MOV      #RUFF,#6
```

T62 TEST FOR PROPER PC ON STACK ON A T HIT TRAP

```
1754 006556 012737 006010 000014      MOV    RPTCT1,INTVEC
1755 006564 012746 000020      MOV    R20,=(6)          ;PUSH 'T' BIT ON THE STACK
1756 006570 012746 006570      MOV    R,+(6)           ;PUSH PC ON THE STACK
1757 006574 000002      RTI                    ;SET 'T' BIT
1758 006570
1759
1760 006576 005212      INC    (R2)             ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1761 006600 012713 000110      MOV    R11,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1762 006604 000000      HLT
1763
1764 006600 000010      LOOP
1765
1766
1767 006610 022737 006570 021770 000000  RPTCT1: CMP    RPTCT1,RUFF+4
1768 006616 001405      HFU    10              ;SCOPE LOOP AND HIT CONTINUE
1769
1770
1771 006620 005212      INC    (R2)             ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1772 006622 012713 000110      MOV    R11,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1773 006626 000000      HLT                    ;CORRECT PC WAS NOT SAVED ON STACK
1774
1775
1776
1777 006630 000010      LOOP
1778
1779
1800 006632 022737 000020 021772 100:  CMP    R20,RUFF-2
1801 006640 001405      HFU    20              ;CORRECT OLD PSW SAVED ON STACK?
1802
1803
1804 006642 005212      INC    (R2)             ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1805 006644 012713 000110      MOV    R11,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1806 006650 000000      HLT                    ;OLD PSW WAS NOT SAVED ON THE STACK
1807
1808
1809
1810 006652 000010      LOOP
1811
1812
1813
1814 006654 010700      20:  SCOPE
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834 006712 005212      INC    (R2)             ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1835 006714 012713 000110      MOV    R11,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1836 006720 000000      HLT                    ;ERROR! 'T' BIT DID NOT TRAP
1837 006722 000010      LOOP                    ;REPLACE THIS INSTRUCTION BY A
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
```

T63 TEST FOR PROPER PC ON THE STACK, ON A T HIT TRAP (PTT)

```
1810 006724 022737 006712 021770 000000  RPTCT1: CMP    RPTCT1,RUFF+4
1811 006732 001405      HFU    10              ;PROPER PC ON THE STACK
1812
1813
1814 006734 005212      INC    (R2)             ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1815 006736 012713 000110      MOV    R11,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1816 006742 000000      HLT                    ;ERROR! IMPROPER PC ON THE STACK
1817 006744 000010      LOOP                    ;REPLACE THIS INSTRUCTION BY A
1818
1819 006746 005720      10:  TST    R3
1820
1821 006750 001005      RNE    20
1822
1823 006752 005212      INC    (R2)             ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1824 006754 012713 000110      MOV    R11,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1825 006760 000000      HLT                    ;T BIT DID NOT ALLOW ONE
1826
1827 006762 000010      LOOP                    ;INSIR AFTER RTT BEFORE
1828
1829
1830
1831 006764
1832
1833
1834 006764 012737 000010 000014      MOV    R10,INTVEC
1835 006772 005037 000010      CLF    TPTVEC+2        ;RESTORE 'T' TRAP
1836 006776 010700      SCOPE                  ;TO HALT AT 10
```

```
1837  
1838  
1839  
1840  
1841 007007 005237 000304 TST64: INC #RTSTN  
1842 007004 012706 000376 MOV #376,R6 ;SET STACK POINTER LESS THAN 400  
1843 007012 012737 007034 000004 MOV #TDEC1,ERRVEC ;LOAD TRAP VECTOR  
1844 007016 005037 000006 CLR #ERRVEC+7  
1845 007022 005710 TST (6) ;DATA FROM LOC 376  
1846 007024 000405 RP TDEC1A  
1847  
1848 007026 005212 INC (R7) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
1849 007030 012713 000116 MOV #116,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
1850 007034 000000 TDEC1: HLT ;ERROR: STACK OVERFLOW TRAP OCCURRED  
1851 ;ON A DATI TO LOC LESS THAN 400  
1852 007036 000240 LOOP ;REPLACE THIS INSTRUCTION BY A  
1853 ;"JMP BR0" (000110) TO GET A  
1854 ;SCOPE LOOP AND HIT CONTINUE  
1855 007040 010700 TDEC1A: SCOPE  
1856  
1857  
1858 ;*****  
1859 ;TEST 65 TEST STACK OVERFLOW TRAP ON A DATIP/DATO TO A STACK LOC. LESS THAN 400  
1860 ;*****  
1861 007042 005237 000304 TST65: INC #RTSTN  
1862 007046 012706 000376 MOV #376,R6  
1863 007052 012737 007036 000004 MOV #TDEC7,ERRVEC ;DATIP/DATO TO BE 376  
1864 007062 000240 CLR (6)  
1865 RP  
1866 007064 005212 INC (R7) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
1867 007066 012713 000117 MOV #117,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
1868 007072 000000 HLT ;PROPR: NO STACK OVERFLOW TRAP (YELLOW)  
1869 007076 000240 LOOP ;REPLACE THIS INSTRUCTION BY A  
1870 ;"JMP BR0" (000110) TO GET A  
1871 ;SCOPE LOOP AND HIT CONTINUE  
1872 007076 010700 TDEC2: SCOPE  
1873  
1874  
1875 ;*****  
1876 ;TEST 66 TEST THAT A DATIP/DATOR CAUSES AN OVERFLOW TRAP  
1877 ;*****  
1878 007100 005237 000304 TST66: INC #RTSTN  
1879 007104 012706 000376 MOV #376,R6  
1880 007110 012737 007134 000004 MOV #TDEC2A,ERRVEC  
1881 007116 012716 177777 R15B #-1,(6)  
1882  
1883 007122 005212 INC (R7) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
1884 007124 012713 000120 MOV #120,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
1885 007130 000000 HLT ;ERROR: NO STACK OVERFLOW TRAP (YELLOW ZONE)  
1886 007132 000240 LOOP ;REPLACE THIS INSTRUCTION BY A  
1887 ;"JMP BR0" (000110) TO GET A  
1888 007134 010700 TDEC2A: SCOPE  
1889  
1890  
1891 ;*****  
1892 ;TEST 67 TEST NO STACK OVERFLOW TRAP ON A DATI (BYT) TO STACK LOC. LESS THAN 4  
1893 ;*****
```

```
1893 007136 005237 000304 TST67: INC #RTSTN  
1894 007142 012705 001400 MOV #1400,R5  
1895 007146 012706 000376 MOV #376,R6  
1896 007152 012737 007172 000004 MOV #TDEC3,ERRVEC  
1897 007160 124645 CNPR -(5),-(6)  
1898 007162 000405 RP TDEC6  
1899  
1900 007164 005212 INC (R7) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
1901 007166 012713 000121 MOV #121,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
1902 007172 000000 TDEC3: HLT ;ERROR: STACK OVERFLOW TRAP OCCURRED  
1903 ;ON A DATI TO STACK LOC UNDER 400  
1904 007174 000240 LOOP ;REPLACE THIS INSTRUCTION BY A  
1905 ;"JMP BR0" (000110) TO GET A  
1906 ;SCOPE LOOP AND HIT CONTINUE  
1907 007176 010700 TDEC6: SCOPE  
1908  
1909 007200 012706 000400 MOV #400,R6  
1910 007204 012737 007224 000004 MOV #TDEC4,ERRVEC  
1911 007212 134546 R15B -(5),-(6)  
1912 007214 000405 RP TDEC6A  
1913  
1914 007216 005212 INC (R7) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
1915 007220 012713 000122 MOV #122,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
1916 007224 000000 TDEC4: HLT ;ERROR: STACK OVERFLOW TRAP OCCURRED  
1917 ;ON A DATI TO STACK LOC UNDER 400  
1918 007226 000240 LOOP ;REPLACE THIS INSTRUCTION BY A  
1919 ;"JMP BR0" (000110) TO GET A  
1920 ;SCOPE LOOP AND HIT CONTINUE  
1921 007230 010700 TDEC6A: SCOPE  
1922  
1923  
1924 ;*****  
1925 ;TEST 70 TEST THAT OVERFLOW TRAP DOES NOT LOSE INFORMATION (OLD PC & PS)  
1926 ;*****  
1927 007232 005237 000304 TST70: INC #RTSTN  
1928 007236 012706 000400 MOV #400,R6  
1929 007242 005037 000376 CLR 376 ;STATUS WORD OF LOC 10  
1930 007246 005037 000374 CLR 374  
1931 007252 012737 007304 000004 MOV #TDEC5,ERRVEC ;RETURN TO LOC 4  
1932 007260 012737 000017 177776 MOV #17,CC ;PRE SET STATUS  
1933 007262 005246 INC -(6)  
1934  
1935 007270 005212 INC (R7) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
1936 007272 012713 000123 MOV #123,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
1937 007276 000000 TDEC5A: HLT ;ERROR: NO STACK OVERFLOW TRAP (YELLOW)  
1938 007300 000240 LOOP ;REPLACE THIS INSTRUCTION BY A  
1939 ;"JMP BR0" (000110) TO GET A  
1940 ;SCOPE LOOP AND HIT CONTINUE  
1941 007302 000415 RP TDEC5B  
1942 007304 022737 000001 000376 CMP #1,376 ;GO TO SCOPE  
1943 007312 000405 BEQ 10 ;WAS INC -(4) EXECUTED  
1944  
1945 007314 005212 INC (R7) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
1946 007316 012713 000124 MOV #124,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
1947 007322 000000 HLT ;ERROR: INSTRUCTION CAUSING YELLOW  
1948 ;ZONE VIOLATION WAS ABORTED. IT  
1949 ;SHOULD HAVE BEEN EXECUTED BEFORE
```

```
1949  
1950 007324 000240 LOOP ;TRAPPING  
1951 ;REPLACE THIS INSTRUCTION BY A  
1952 ;"JMP BR0" (000110) TO GET A  
1953 007326 022737 000001 000374 101 CMP #1,374 ;SCOPE LOOP AND HIT CONTINUE  
1954 ;WAS STATUS SAVED NOTE: INC DOES NOT  
1955 007334 001400 HND TOECSA ;AFFECT 'C' BIT IN STATUS.  
1956 007336 010700 ;DECSR: SCOPE  
1957  
1958  
1959 ;*****  
1960 ;*TEST 71 TEST THAT A RESERVED INST CAUSES AN OVERFLOW TRAP  
1961 ;*****  
1962 007340 005237 000304 TST71: INC #0TESTN  
1963 007344 012706 000400 MOV #400,36 ;SET UP STACK TO OVERFLOW  
1964 007350 012737 007404 000014 MOV #VDEC7,10 ;SET UP INST VECTOR  
1965 007356 012737 007410 000004 MOV #VDEC1,4 ;SET UP OVERFLOW VECTOR  
1966 007364 075040 75040 ;THIS TRAP SHOULD CAUSE OVERFLOW  
1967  
1968 007366 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
1969 007370 012713 000125 MOV #125,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
1970 007374 000000 HLT ;NO TRAP OCCURRED  
1971  
1972 007376 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
1973 007400 012713 000126 MOV #126,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
1974 007404 000000 HLT ;TRAP FLAG OVERFLOW DID NOT OCCUR  
1975 007406 000240 LOOP ;REPLACE THIS INSTRUCTION BY A  
1976 ;"JMP BR0" (000110) TO GET A  
1977 ;SCOPE LOOP AND HIT CONTINUE  
1978 007410 010700 VDEC1: SCOPE ;NORMAL OVERFLOW RETURN  
1979  
1980 ;*****  
1981 ;*TEST 72 TEST THAT AN "IOT" CAUSES AN OVERFLOW TRAP  
1982 ;*****  
1983 007412 005237 000304 TST72: INC #0TESTN  
1984 007416 012706 000400 MOV #400,36 ;SET UP STACK TO OVERFLOW  
1985 007422 012737 007456 000020 MOV #VDEC4,20 ;SET UP INST VECTOR  
1986 007430 012737 007462 000004 MOV #VDEC1,4 ;SET UP OVERFLOW VECTOR  
1987 007436 000004 IOT ;THIS TRAP SHOULD CAUSE OVERFLOW  
1988  
1989 007440 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
1990 007442 012713 000127 MOV #127,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
1991 007446 000000 HLT ;NO TRAP OCCURRED  
1992  
1993 007450 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
1994 007452 012713 000130 MOV #130,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
1995 007456 000000 HLT ;TRAP FLAG OVERFLOW DID NOT OCCUR  
1996 007460 000240 LOOP ;REPLACE THIS INSTRUCTION BY A  
1997 ;"JMP BR0" (000110) TO GET A  
1998 ;SCOPE LOOP AND HIT CONTINUE  
1999 007462 010700 VDEC3: SCOPE ;NORMAL OVERFLOW RETURN  
2000  
2001 ;*****  
2002 ;*TEST 73 TEST THAT AN "EMT" CAUSES AN OVERFLOW TRAP  
2003 ;*****  
2004 007464 005237 000304 TST73: INC #0TESTN  
2005 007470 012706 000400 MOV #400,36 ;SET UP STACK TO OVERFLOW
```

```
2006 007502 012737 007530 000030 MOV #VDEC6,10 ;SET UP INST VECTOR  
2007 007510 104000 007534 000004 MOV #VDEC5,4 ;SET UP OVERFLOW VECTOR  
2008 ;THIS TRAP SHOULD CAUSE OVERFLOW  
2009  
2010 007512 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
2011 007514 012713 000131 MOV #131,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
2012 007520 000000 HLT ;NO TRAP OCCURRED  
2013  
2014 007522 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
2015 007524 012713 000132 MOV #132,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
2016 007530 000000 HLT ;TRAP FLAG OVERFLOW DID NOT OCCUR  
2017 007532 000240 LOOP ;REPLACE THIS INSTRUCTION BY A  
2018 ;"JMP BR0" (000110) TO GET A  
2019 007534 010700 VDEC5: SCOPE ;SCOPE LOOP AND HIT CONTINUE  
2020 ;NORMAL OVERFLOW RETURN  
2021  
2022 ;*****  
2023 ;*TEST 74 TEST THAT AN "TRAP" CAUSES AN OVERFLOW TRAP  
2024 ;*****  
2025 007536 005237 000304 TST74: INC #0TESTN  
2026 007542 012706 000400 MOV #400,36 ;SET UP STACK TO OVERFLOW  
2027 007546 012737 007602 000034 MOV #VDEC8,34 ;SET UP INST VECTOR  
2028 007554 012737 007606 000004 MOV #VDEC7,4 ;SET UP OVERFLOW VECTOR  
2029 007562 104000 TRAP ;THIS TRAP SHOULD CAUSE OVERFLOW  
2030  
2031 007564 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
2032 007566 012713 000133 MOV #133,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
2033 007572 000000 HLT ;NO TRAP OCCURRED  
2034  
2035 007574 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
2036 007576 012713 000134 MOV #134,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
2037 007602 000000 HLT ;TRAP FLAG OVERFLOW DID NOT OCCUR  
2038 007604 000240 LOOP ;REPLACE THIS INSTRUCTION BY A  
2039 ;"JMP BR0" (000110) TO GET A  
2040 007606 010700 VDEC7: SCOPE ;SCOPE LOOP AND HIT CONTINUE  
2041 ;NORMAL OVERFLOW RETURN  
2042  
2043 ;*****  
2044 ;*TEST 75 TEST THAT AN BPT CAUSES AN OVERFLOW TRAP  
2045 ;*****  
2046 007610 005237 000304 TST75: INC #0TESTN  
2047 007614 012706 000400 MOV #400,36 ;SET UP STACK TO OVERFLOW  
2048 007620 012737 007654 000014 MOV #VDEC10,14 ;SET UP INST VECTOR  
2049 007626 012737 007660 000004 MOV #VDEC9,4 ;SET UP OVERFLOW VECTOR  
2050 007634 000003 BPT ;THIS TRAP SHOULD CAUSE OVERFLOW  
2051  
2052 007636 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
2053 007640 012713 000135 MOV #135,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
2054 007644 000000 HLT ;NO TRAP OCCURRED  
2055  
2056 007646 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
2057 007650 012713 000136 MOV #136,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
2058 007654 000000 HLT ;TRAP FLAG OVERFLOW DID NOT OCCUR  
2059 007656 000240 LOOP ;REPLACE THIS INSTRUCTION BY A  
2060 ;"JMP BR0" (000110) TO GET A  
;SCOPE LOOP AND HIT CONTINUE
```

```
2061 007060 010700 VDEC9: SCOPE ;NORMAL OVERFLOW RETURN
2062
2063 ;*****
2064 ;*TEST 76 TEST THAT AN ILLEGAL INSTP. CAUSES AN OVERFLOW TRAP
2065 ;*****
2066 007662 005237 000304 TST76: INC 000TESTN
2067 007666 012706 000400 MOV 0400,06 ;SET UP STACK TO OVERFLOW
2068 007672 012737 000004 000004 MOV 0VDEC12,4 ;SET UP INST VECTOR
2069 007700 012737 000732 000004 MOV 0VDEC11,4 ;SET UP OVERFLOW VECTOR
2070 007706 004700 HLLA ;THIS TRAP SHOULD CAUSE OVERFLOW
2071
2072 007710 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2073 007712 012713 000137 MOV 0137,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2074 007716 000000 HLT ;NO TRAP OCCURRED
2075
2076 007720 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2077 007722 012713 000140 MOV 0140,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2078 007726 000000 HLT ;TRAP FLAG OVERFLOW DID NOT OCCUR
2079 007730 000740 LOOP ;REPLACE THIS INSTRUCTION BY A
2080 ;"JMP 000" (000110) TO GET A
2081 ;SCOPE LOOP AND HIT CONTINUE
2082 007732 010700 VDEC11: SCOPE ;NORMAL OVERFLOW RETURN
2083 007734 020627 000370 CMP 06,0370 ;STACK PUSHED FOUR WORDS?
2084 007740 001405 BEQ 16
2085
2086 007742 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2087 007744 012713 000141 MOV 0141,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2088 007750 000000 HLT ;TRAP OVERFLOW DID NOT OCCUR
2089 007752 000740 LOOP ;REPLACE THIS INSTRUCTION BY A
2090 ;"JMP 000" (000110) TO GET A
2091 ;SCOPE LOOP AND HIT CONTINUE
2092 007754 010700 16: SCOPE
2093
2094 ;*****
2095 ;*TEST 77 TEST THAT AN ILLEGAL INSTP. CAUSES AN OVERFLOW TRAP
2096 ;*****
2097 007756 005237 000304 TST77: INC 000TESTN
2098 007762 012706 000400 MOV 0400,06 ;SET UP STACK TO OVERFLOW
2099 007766 012737 010022 000004 000004 MOV 0VDEC14,4 ;SET UP INST VECTOR
2100 007774 012737 010026 000004 MOV 0VDEC13,4 ;SET UP OVERFLOW VECTOR
2101 010002 000100 HLLA ;THIS TRAP SHOULD CAUSE OVERFLOW
2102
2103 010004 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2104 010006 012713 000142 MOV 0142,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2105 010012 000000 HLT ;NO TRAP OCCURRED
2106
2107 010014 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2108 010016 012713 000143 MOV 0143,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2109 010022 000000 HLT ;TRAP FLAG OVERFLOW DID NOT OCCUR
2110 010024 000740 LOOP ;REPLACE THIS INSTRUCTION BY A
2111 ;"JMP 000" (000110) TO GET A
2112 ;SCOPE LOOP AND HIT CONTINUE
2113 010026 010700 VDEC13: SCOPE ;NORMAL OVERFLOW RETURN
2114
2115 ;INSTRUCTION EQUATE STATEMENTS
2116
```

```
2117 ; 4510 #JSP S,(0)
2118 ; 005040 #CLR -(6)
2119 ; 010046 #MOV 00,-(6)
2120 ; 006746 #SXT -(6)
2121 ; 074046 #XOR 00,-(6)
2122
2123 ;*****
2124 ;*TEST 100 TEST THAT THE INSTRUCTION (4510) CAUSES STACK OVERFLOW CONDITION
2125 ;*****
2126 010030 005237 000304 TST100: INC 000TESTN
2127 010034 012706 000400 MOV 0400,06 ;SET STACK POINTER
2128 010040 005000 CLR 00 ;PRE SET R0
2129 010042 012737 010064 000004 MOV 0VDEC15,4 ;LOAD ERROR VECTOR
2130 010050 004510 4510 ;CAUSE OVERFLOW
2131
2132 010052 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2133 010054 012713 000144 MOV 0144,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2134 010060 000000 HLT ;ERROR! OVERFLOW FAILED TO TRAP
2135 010062 000740 LOOP ;REPLACE THIS INSTRUCTION BY 15
2136 ;"JMP 000" (000110) TO GET 15
2137 ;SCOPE LOOP AND HIT CONTINUE
2138 010064 022706 000372 VDEC15: CMP 0372,06 ;HAS STACK POINTER MOVED BY 6
2139 010070 001405 BFG 16 ;12 FOR THE AUTO DECREMENT + 4 FOR
2140
2141 010072 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2142 010074 012713 000145 MOV 0145,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2143 010100 000000 HLT ;THE ERROR TRAP
2144 ;R6 DID NOT DECREMENT BY 6 WHEN
2145 ;AN INSTRUCTION DOING AUTO-DECREMENT
2146 ;OF R6 CAUSED YELLOW ZONE VIOLATION
2147 010102 000740 LOOP ;REPLACE THIS INSTRUCTION BY 15
2148 ;"JMP 000" (000110) TO GET 15
2149 ;SCOPE LOOP AND HIT CONTINUE
2150 010104 010700 16: SCOPE
2151
2152 ;*****
2153 ;*TEST 101 TEST THAT THE INSTRUCTION (005046) CAUSES STACK OVERFLOW CONDITION
2154 ;*****
2155 010106 005237 000304 TST101: INC 000TESTN
2156 010112 012706 000400 MOV 0400,06 ;SET STACK POINTER
2157 010116 005000 CLR 00 ;PRE SET R0
2158 010120 012737 010142 000004 000004 MOV 0VDEC16,4 ;LOAD ERROR VECTOR
2159 010126 005046 005046 ;CAUSE OVERFLOW
2160
2161 010130 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2162 010132 012713 000146 MOV 0146,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2163 010136 000000 HLT ;ERROR! OVERFLOW FAILED TO TRAP
2164 010140 000740 LOOP ;REPLACE THIS INSTRUCTION BY 16
2165 ;"JMP 000" (000110) TO GET 16
2166 ;SCOPE LOOP AND HIT CONTINUE
2167 010142 022706 000372 VDEC16: CMP 0372,06 ;HAS STACK POINTER MOVED BY 6
2168 010146 001405 BEQ 16 ;12 FOR THE AUTO DECREMENT + 4 FOR
2169
2170 010150 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2171 010152 012713 000147 MOV 0147,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2172 010156 000000 HLT ;THE ERROR TRAP
```

```
2173 ;R6 DID NOT DECREMENT BY 6 WHEN
2174 ;AN INSTRUCTION DOING AUTO-DECREMENT
2175 ;OF R6 CAUSED YELLOW ZONE VIOLATION
2176 010160 000240 LOOP ;REPLACE THIS INSTRUCTION BY 10
2177 ;"JMP RRW" (000110) TO GET 10
2178 ;SCOPE LOOP AND HIT CONTINUE
2179
2180
2181 ;*****
2182 ;*TEST 102 TEST THAT THE INSTRUCTION (010046) CAUSES STACK OVERFLOW CONDITION
2183 ;*****
2184 010164 005237 000304 TST102: INC 001TEST4
2185 010170 012706 000400 MOV 0400,R6 ;SET STACK POINTER
2186 010174 005400 CLR R0 ;PRE SET R0
2187 010176 012737 010220 000004 MOV 0VDEC17,4 ;LOAD ERROR VECTOR
2188 010204 010046 ;CAUSE OVERFLOW
2189
2190
2191 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2192 MOV #150,(R3) ;MOVE TO MAIL BOX (FATAL) THE ERROR
2193 HLT ;ERROR: OVERFLOW FAILED TO TRAP
2194 ;REPLACE THIS INSTRUCTION BY 17
2195 ;"JMP RRW" (000110) TO GET 17
2196 ;SCOPE LOOP AND HIT CONTINUE
2197 010120 022706 000372 VDEC17: CMP 0372,R6
2198 010224 001405 BEQ 10 ;HAS STACK POINTER MOVED BY 6
2199 ;(2 FOR THE AUTO DECREMENT + 4 FOR
2200
2201
2202 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2203 MOV #151,(R3) ;MOVE TO MAIL BOX (FATAL) THE ERROR
2204 HLT ;THE ERROR TRAP)
2205 ;R6 DID NOT DECREMENT BY 6 WHEN
2206 ;AN INSTRUCTION DOING AUTO-DECREMENT
2207 ;OF R6 CAUSED YELLOW ZONE VIOLATION
2208 ;REPLACE THIS INSTRUCTION BY 17
2209 ;"JMP RRW" (000110) TO GET 17
2210 ;SCOPE LOOP AND HIT CONTINUE
2211
2212 ;*****
2213 ;*TEST 103 TEST THAT THE INSTRUCTION (000746) CAUSES STACK OVERFLOW CONDITION
2214 ;*****
2215 010242 005237 000304 TST103: INC 001TEST4
2216 010246 012736 000400 MOV 0400,R6 ;SET STACK POINTER
2217 010252 005000 CLR R0 ;PRE SET R0
2218 010254 012737 010276 000004 MOV 0VDEC18,4 ;LOAD ERROR VECTOR
2219 010262 006746 000746 ;CAUSE OVERFLOW
2220
2221
2222 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2223 MOV #152,(R3) ;MOVE TO MAIL BOX (FATAL) THE ERROR
2224 HLT ;ERROR: OVERFLOW FAILED TO TRAP
2225 ;REPLACE THIS INSTRUCTION BY 10
2226 ;"JMP RRW" (000110) TO GET 10
2227 ;SCOPE LOOP AND HIT CONTINUE
2228 010276 022706 000372 VDEC18: CMP 0372,R6
2229 010302 001405 BEQ 10 ;HAS STACK POINTER MOVED BY 6
2230 ;(2 FOR THE AUTO DECREMENT + 4 FOR
2231
2232
2233 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
```

```
2229 010306 012753 000153 MOV #153,(R3) ;MOVE TO MAIL BOX (FATAL) THE ERROR
2230 010312 000000 HLT ;THE ERROR TRAP)
2231 ;R6 DID NOT DECREMENT BY 6 WHEN
2232 ;AN INSTRUCTION DOING AUTO-DECREMENT
2233 ;OF R6 CAUSED YELLOW ZONE VIOLATION
2234 010311 000746 LOOP ;REPLACE THIS INSTRUCTION BY 10
2235 ;"JMP RRW" (000110) TO GET 10
2236 ;SCOPE LOOP AND HIT CONTINUE
2237
2238
2239 ;*****
2240 ;*TEST 104 TEST THAT THE INSTRUCTION (070046) CAUSES STACK OVERFLOW CONDITION
2241 ;*****
2242 010320 005237 000304 TST104: INC 001TEST4
2243 010324 012706 000400 MOV 0400,R6 ;SET STACK POINTER
2244 010330 005000 CLR R0 ;PRE SET R0
2245 010332 012737 010354 000004 MOV 0VDEC19,4 ;LOAD ERROR VECTOR
2246 010340 070046 070046 ;CAUSE OVERFLOW
2247
2248
2249 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2250 MOV #154,(R3) ;MOVE TO MAIL BOX (FATAL) THE ERROR
2251 HLT ;ERROR: OVERFLOW FAILED TO TRAP
2252 ;REPLACE THIS INSTRUCTION BY 19
2253 ;"JMP RRW" (000110) TO GET 19
2254 ;SCOPE LOOP AND HIT CONTINUE
2255 010354 022706 000372 VDEC19: CMP 0372,R6
2256 010360 001405 BEQ 10 ;HAS STACK POINTER MOVED BY 6
2257 ;(2 FOR THE AUTO DECREMENT + 4 FOR
2258
2259
2260 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2261 MOV #155,(R3) ;MOVE TO MAIL BOX (FATAL) THE ERROR
2262 HLT ;THE ERROR TRAP)
2263 ;R6 DID NOT DECREMENT BY 6 WHEN
2264 ;AN INSTRUCTION DOING AUTO-DECREMENT
2265 ;OF R6 CAUSED YELLOW ZONE VIOLATION
2266 ;REPLACE THIS INSTRUCTION BY 19
2267 ;"JMP RRW" (000110) TO GET 19
2268 ;SCOPE LOOP AND HIT CONTINUE
```



```
2269
2270
2271 ;*****
2272 ;*TEST 106 TEST FOR FALSE OVERFLOW TRAP
2273 ;*****
2274 ;*****
2275 ;PROGRAM MAY HAVE RELOADED IF OVERFLOW FAILS
2276 010402 012717 010500 000004 MOV 0402,06 ;SET UP OVERFLOW POINTER
2277 010410 012706 000402 MOV 0402,06
2278 010414 010605 MOV 06,05
2279 010416 005746 TST =0 ;SHOULD NOT OVERFLOW
2280 010420 012706 001002 MOV 01002,06
2281 010424 010605 MOV 06,05
2282 010426 005746 TST =0 ;SHOULD NOT OVERFLOW
2283 010430 012706 002002 MOV 02002,06
2284 010434 010605 MOV 06,05
2285 010436 005746 TST =0 ;SHOULD NOT OVERFLOW
2286 010440 012706 004002 MOV 04002,06
2287 010444 010605 MOV 06,05
2288 010446 005746 TST =0 ;SHOULD NOT OVERFLOW
2289 010450 012706 010002 MOV 010002,06
2290 010454 010605 MOV 06,05
2291 010456 005746 TST =0 ;SHOULD NOT OVERFLOW
2292 010460 012706 020000 MOV 020000,06
2293 010464 010605 MOV 06,05
2294 010466 005746 TST =0
2295 010470 000000 RP 0001
2296
2297 010472 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2298 010474 012713 000156 MOV 0156,(R1) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2299 010500 000000 FOVER: HLT ;FALSE STACK OVERFLOW TRAP OCCURRED
2300 ;ON DOING AN AUTO-DECREMENT OF R6
2301 ;THE INITIAL R6 IS IN SAVED IN R5
2302 ;CHECK STACK TO FIND WHERE
2303 ;REPLACE THIS INSTRUCTION BY A
2304 ;"JMP 000" (000110) TO GET A
2305 ;SCOPE LOOP AND HIT CONTINUE
2306 010504 010700 FOV1: SCOPE
2307
2308 ;*****
2309 ;*TEST 106 TEST FOR REPEATED TRAPS AND OVERFLOW OF STACK
2310 ;*****
2311 010506 005237 000104 IST106: INC 005237M
2312 ;WHAT THE STACK LOOKS LIKE WHEN INSTRUCTION AT "04" IS BEING EXECUTED
2313 ; OVFLW PC=05,STAT=0 04 342 BOTTOM OF STACK
2314 ; 4 344
2315 ; TRT PC=04,STAT=4 03A 346
2316 ; 0 350
2317 ; OVFLW PC=05,STAT=0 03 352
2318 ; 3 354
2319 ; IOT PC=03,STAT=3 02A 356
2320 ; 0 360
2321 ; OVFLW PC=05,STAT=0 02 362
2322 ; 2 364
2323 ; TRAP PC=02,STAT=2 01A 366
2324 ; 0 370
```

```
2325 ; OVFLW PC=05,STAT=0 01 372
2326 ; 1 374
2327 ; ENT PC=01,STAT=1 00A 376
2328 ; 17 400
2329 ;
2330 010512 012706 000402 MOV 0402,06 ;INITIALIZE STACK POINTER
2331 010516 012717 010032 000030 MOV 001,0 ;NEW PC POINTS TO NEXT INSTP.
2332 010524 012717 000001 000032 MOV 01,0+1 ;NEW PSW
2333 010532 012717 010646 000034 MOV 002,TRAPVEC ;NEW PC POINTS TO NEXT INSTR
2334 010540 012717 000002 000036 MOV 02,TRAPVEC+2 ;NEW PSW
2335 010544 012717 010662 000020 MOV 003,IOTVEC ;NEW PC POINTS TO NEXT INSTR
2336 010554 012717 000003 000022 MOV 03,IOTVEC+2 ;NEW PSW
2337 010562 012717 010676 000014 MOV 004,TRTVEC ;NEW PC POINTS TO NEXT INSTR
2338 010574 012717 000004 000016 MOV 04,TRTVEC+2 ;NEW PSW
2339 010576 012717 011324 000004 MOV 005,ERRVEC ;NEW PC POINTS TO NEXT INSTR
2340 010604 005037 000000 CLP 000000 ;NEW PSW
2341 010610 012717 000017 177776 MOV 017,CC ;PRESENT PSW
2342 010616 000000 OV: ENT
2343 010622 000000 OVI:
2344
2345 010620 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2346 010622 012713 000157 MOV 0157,(R1) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2347 010626 000000 OVA: HLT ;EMI NOT EXECUTED
2348 010630 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
2349 ;"JMP 000" (000110) TO GET A
2350 ;SCOPE LOOP AND HIT CONTINUE
2351 010632 004400 O1: TRAP ;RETURN HERE FROM PREVIOUS YELLOW ZONE
2352 ;STACK VIOLATION
2353 010634 000000 O11:
2354
2355 010634 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2356 010636 012713 000160 MOV 0160,(R1) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2357 010642 000000 O1A: HLT ;IOT NOT EXECUTED
2358 010644 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
2359 ;"JMP 000" (000110) TO GET A
2360 ;SCOPE LOOP AND HIT CONTINUE
2361 010646 000000 O2: IOT ;RETURN HERE FROM PREVIOUS YELLOW ZONE
2362 ;STACK VIOLATION
2363 010650 000000 O21:
2364
2365 010654 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2366 010656 012713 000161 MOV 0161,(R1) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2367 010662 000000 O2A: HLT ;IOT NOT EXECUTED
2368 010664 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
2369 ;"JMP 000" (000110) TO GET A
2370 ;SCOPE LOOP AND HIT CONTINUE
2371 010666 000000 O3: BPT ;RETURN HERE FROM PREVIOUS YELLOW ZONE
2372 ;STACK VIOLATION
2373 010668 000000 O31:
2374
2375 010664 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2376 010666 012713 000162 MOV 0162,(R1) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2377 010672 000000 O3A: HLT ;BPT NOT EXECUTED
2378 010674 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
2379 ;"JMP 000" (000110) TO GET A
2380 ;SCOPE LOOP AND HIT CONTINUE
```

T106 TEST FOR REPEATED TRAPS AND OVERFLOW OF STACK

2381	010676	022700	000342	04:	CMP	0342,00	;	IS STACK POINTER
2382	010702	001405			BEQ	200	;	POSITIONED PROPERLY
2383								
2384	010704	005212			INC	(R2)	;	SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2385	010706	012713	000103		MOV	#163,(R3)	;	MOVE TO MAIL BOX (\$FATAL) THE ERROR
2386	010712	000000			HLT		;	ERROR! INCONRECT STACK POINTER AFTER
2387							;	4 REPETITIVE STACK OVERFLOW TRAPS
2388	010714	000240			LOOP		;	REPLACE THIS INSTRUCTION BY A
2389							;	"JMP 000" (000110) TO GET A
2390							;	SCOPE LOOP AND HIT CONTINUE
2391	010716	012720	000402	20:	MOV	#002,00		
2392	010722	022740	000017		CMP	#17,-(R)	;	CORRECT OLD PSW SAVED?
2393	010726	001405			REQ	15		
2394								
2395	010730	005212			INC	(R2)	;	SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2396	010732	012713	000104		MOV	#164,(R3)	;	MOVE TO MAIL BOX (\$FATAL) THE ERROR
2397	010736	000000			HLT		;	CORRECT PSW WAS NOT SAVED
2398							;	ON EMT TRAP
2399	010740	000240			LOOP		;	REPLACE THIS INSTRUCTION BY A
2400							;	"JMP 000" (000110) TO GET A
2401							;	SCOPE LOOP AND HIT CONTINUE
2402	010742	022740	010020	10:	CMP	0001,-(R)		
2403	010746	001405			REQ	20		
2404								
2405	010750	005212			INC	(R2)	;	SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2406	010752	012713	000105		MOV	#105,(R3)	;	MOVE TO MAIL BOX (\$FATAL) THE ERROR
2407	010756	000000			HLT		;	CORRECT PC WAS NOT SAVED ON
2408							;	EMT TRAP
2409	010760	000240			LOOP		;	REPLACE THIS INSTRUCTION BY A
2410							;	"JMP 000" (000110) TO GET A
2411							;	SCOPE LOOP AND HIT CONTINUE
2412	010762	022740	000001	20:	CMP	#1,-(R)		
2413	010766	001405			REQ	30		
2414								
2415	010770	005212			INC	(R2)	;	SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2416	010772	012713	000106		MOV	#166,(R3)	;	MOVE TO MAIL BOX (\$FATAL) THE ERROR
2417	010776	000000			HLT		;	CORRECT PSW WAS NOT SAVED ON STACK
2418							;	OVERFLOW TRAP FOLLOWING EMT
2419	010780	000240			LOOP		;	REPLACE THIS INSTRUCTION BY A
2420							;	"JMP 000" (000110) TO GET A
2421							;	SCOPE LOOP AND HIT CONTINUE
2422	011002	022710	010032	30:	CMP	#01,-(R)		
2423	011006	001405			REQ	40		
2424								
2425	011010	005212			INC	(R2)	;	SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2426	011012	012713	000167		MOV	#167,(R3)	;	MOVE TO MAIL BOX (\$FATAL) THE ERROR
2427	011016	000000			HLT		;	CORRECT PC WAS NOT SAVED ON STACK
2428							;	OVERFLOW TRAP FOLLOWING EMT
2429	011020	000240			LOOP		;	REPLACE THIS INSTRUCTION BY A
2430							;	"JMP 000" (000110) TO GET A
2431							;	SCOPE LOOP AND HIT CONTINUE
2432	011022	022740	000000	40:	CMP	#0,-(R)		
2433	011026	001405			REQ	50		
2434								
2435	011030	005212			INC	(R2)	;	SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2436	011032	012713	000170		MOV	#170,(R3)	;	MOVE TO MAIL BOX (\$FATAL) THE ERROR

T106 TEST FOR REPEATED TRAPS AND OVERFLOW OF STACK

2437	011036	000000			HLT		;	CORRECT PSW WAS NOT SAVED ON
2438							;	"TRAP" TRAP
2439	011040	000240			LOOP		;	REPLACE THIS INSTRUCTION BY A
2440							;	"JMP 000" (000110) TO GET A
2441							;	SCOPE LOOP AND HIT CONTINUE
2442	011042	022740	010034	50:	CMP	#011,-(R)		
2443	011046	001405			REQ	60		
2444								
2445	011050	005212			INC	(R2)	;	SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2446	011052	012713	000171		MOV	#171,(R3)	;	MOVE TO MAIL BOX (\$FATAL) THE ERROR
2447	011056	000000			HLT		;	CORRECT PC WAS NOT SAVED ON
2448							;	"TRAP" TRAP
2449	011060	000240			LOOP		;	REPLACE THIS INSTRUCTION BY A
2450							;	"JMP 000" (000110) TO GET A
2451							;	SCOPE LOOP AND HIT CONTINUE
2452	011062	022740	000002	60:	CMP	#2,-(R)		
2453	011066	001405			REQ	70		
2454								
2455	011070	005212			INC	(R2)	;	SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2456	011072	012713	000172		MOV	#172,(R3)	;	MOVE TO MAIL BOX (\$FATAL) THE ERROR
2457	011076	000000			HLT		;	CORRECT PSW WAS NOT SAVED ON STACK
2458							;	OVERFLOW TRAP FOLLOWING "TRAP"
2459	011100	000240			LOOP		;	REPLACE THIS INSTRUCTION BY A
2460							;	"JMP 000" (000110) TO GET A
2461							;	SCOPE LOOP AND HIT CONTINUE
2462	011102	022740	010046	70:	CMP	#02,-(R)		
2463	011106	001405			REQ	80		
2464								
2465	011110	005212			INC	(R2)	;	SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2466	011112	012713	000173		MOV	#173,(R3)	;	MOVE TO MAIL BOX (\$FATAL) THE ERROR
2467	011116	000000			HLT		;	CORRECT PC WAS NOT SAVED ON STACK
2468							;	OVERFLOW TRAP FOLLOWING "TRAP"
2469	011120	000240			LOOP		;	REPLACE THIS INSTRUCTION BY A
2470							;	"JMP 000" (000110) TO GET A
2471							;	SCOPE LOOP AND HIT CONTINUE
2472	011122	022740	000000	80:	CMP	#0,-(R)		
2473	011126	001405			REQ	90		
2474								
2475	011130	005212			INC	(R2)	;	SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2476	011132	012713	000174		MOV	#174,(R3)	;	MOVE TO MAIL BOX (\$FATAL) THE ERROR
2477	011136	000000			HLT		;	CORRECT PSW WAS NOT SAVED ON
2478							;	"IOT" TRAP
2479	011140	000240			LOOP		;	REPLACE THIS INSTRUCTION BY A
2480							;	"JMP 000" (000110) TO GET A
2481							;	SCOPE LOOP AND HIT CONTINUE
2482	011142	022740	010050	90:	CMP	#021,-(R)		
2483	011146	001405			REQ	100		
2484								
2485	011150	005212			INC	(R2)	;	SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2486	011152	012713	000175		MOV	#175,(R3)	;	MOVE TO MAIL BOX (\$FATAL) THE ERROR
2487	011156	000000			HLT		;	CORRECT PC WAS NOT SAVED ON
2488							;	"IOT" TRAP
2489	011160	000240			LOOP		;	REPLACE THIS INSTRUCTION BY A
2490							;	"JMP 000" (000110) TO GET A
2491							;	SCOPE LOOP AND HIT CONTINUE
2492	011162	022740	000003	100:	CMP	#3,-(R)		



```

2605 011522 000000 HLT ;ERROR: TRT VECTOR (TRTVFC) NOT SAVED
2606 011524 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
2607 ;"JMP BR" (000110) TO GET A
2608 ;SCOPE LOOP AND HIT CONTINUE
2609 011526 000000 BR TPTEXT ;EXIT TEST
2610 011530 000000 RTI ;RETURN FROM OVERFLOW TRAP
2611
2612 011532 010005 TPIC: MOV 00,05 ;GET STACK POINTER
2613 011534 162715 000000 SHL 02,(5) ;POINT PC TO TRAPPED INSTRUCTION
2614 011540 013335 021416 TEMP+2,(5) ;RESUME ORIGINAL INSTRUCTION (COM TEMP)
2615 011544 052715 000000 LIS 020,(5) ;SET "T" BIT ON STACK
2616 011550 012737 011566 000000 MOV #TRTD,TRTVFC ;CHANGE ERROR VECTOR
2617 011556 012737 011702 000000 MOV #ATMB,TRTVFC ;LOAD "T" BIT VECTOR
2618 011564 000000 RTI ;GO DO ORIGINAL INSTRUCTION AND TRAP
2619 ;("T" BIT TRAP) WHEN FINISHED
2620
2621
2622 011566 022737 000000 000376 ;CHECK THE STACK
2623 011574 001405 TRTD: CME #01,376 ;IS "N","C","T" BITS SET ON THE STACK
2624 ;(RESULT OF COM TEMP IN TRTA
2625
2626 011576 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2627 011600 012713 000210 MOV #210,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
2628 011604 000000 HLT ;ERROR: STATUS NOT SAVED
2629 011606 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
2630 ;"JMP BR" (000110) TO GET A
2631 011610 022737 011414 000376 15: CME #TRTA,376 ;WAS RETURN PC FROM "T" TRAP SAVED?
2632 011616 001406 BEQ 25
2633
2634 011620 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2635 011622 012713 000211 MOV #211,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
2636 011626 000000 HLT ;ERROR: RETURN PC NOT SAVED
2637 011630 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
2638 ;"JMP BR" (000110) TO GET A
2639 ;SCOPE LOOP AND HIT CONTINUE
2640
2641 011632 000437 BR TPTEXT ;EXIT TEST
2642 011634 005737 000472 28: TST 372 ;"T" BIT TRAP STATUS SAVED?
2643 011640 001405 BEQ 35
2644
2645 011642 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2646 011644 012713 000212 MOV #212,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
2647 011650 000000 HLT ;ERROR: "T" BIT STATUS (TRTVFC+2) NOT SAVED
2648 011652 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
2649 ;"JMP BR" (000110) TO GET A
2650 ;SCOPE LOOP AND HIT CONTINUE
2651
2652 011654 022737 011702 000376 36: CME #TRTF,376 ;"T" BIT VECTOR SAVED (TRTVFC)
2653 011662 001406 BEQ 48
2654
2655 011664 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2656 011666 012713 000213 MOV #213,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
2657 011672 000000 HLT ;ERROR: "T" BIT VECTOR NOT SAVED
2658 011674 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
2659 ;"JMP BR" (000110) TO GET A
2660 ;SCOPE LOOP AND HIT CONTINUE
2661
2662 011676 000415 BR TPTEXT ;EXIT TEST
2663 011700 000000 RTI ;RETURN FROM ERROR TRAP
  
```

```

2661
2662 011702 005037 000376 TRTD: CLR 376 ;CLEAR SAVED "T" BIT
2663 011706 000000 RTI ;RETURN FROM "T" BIT TRAP
2664
2665 011710 022737 177777 021414 TPTR: CME 00,TEMP ;WAS ORIGINAL INSTRUCTION EXECUTED?
2666 011716 001405 BEQ INTEXT
2667
2668 011720 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2669 011722 012713 000214 MOV #214,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
2670 011726 000000 HLT ;TRAPPED INSTRUCTION (COM TEMP) NOT EXEC.
2671 011730 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
2672 ;"JMP BR" (000110) TO GET A
2673 ;SCOPE LOOP AND HIT CONTINUE
2674 011732 012737 000016 000000 TPICX: MOV #TRTVFC+2,TRTVFC ;RESTORE TRAP
2675 011740 012737 000006 000000 MOV 00,1 ;VECTORS
2676 011746 010700 SCOPE
2677
2678 ;*****
2679 ;TEST 11X TEST YELLOW ZONE STACK VIOLATION, USING INDEX MODE (6) & REL. (R6,GT,4)
2680 ;*****
2681 011750 005237 000304 TSLI1: INC #1,TEXTN
2682 011754 012706 001000 INCEX: MOV #1000,06 ;SET UP STACK POINTER
2683 011760 012737 012010 000000 MOV #1200,4
2684 011766 005037 000000 CLR 6
2685 011772 005060 177340 CLF #40(6) ;FINAL ADDRESS IS 340
2686
2687 011776 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2688 012000 012713 000215 MOV #215,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
2689 012004 000000 HLT ;ERROR: STACK OVERFLOW TRAP DID NOT OCCUR
2690 012006 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
2691 ;"JMP BR" (000110) TO GET A
2692 ;SCOPE LOOP AND HIT CONTINUE
2693
2694 012010 010704 IND1: SCOPE
2695
2696 ;TEST THAT THE TRAP SEQUENCE IS EXECUTED PROPERLY ON OVERFLOW.
2697 012012 012706 001000 MOV #1000,05
2698 012016 012737 012064 000000 MOV #1204,4
2699 012024 005037 000776 CLR 776
2700 012030 005037 000774 CLR 774
2701 012034 012737 000000 000176 MOV #1,376 ;PRE SET DEST ADDRESS
2702 012042 012706 000000 177376 MOV #0,-402(6) ;FINAL ADDRESS IS 376
2703
2704 012050 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2705 012052 012713 000216 MOV #216,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
2706 012056 000000 HLT ;ERROR: FAILED TO TRAP
2707 012060 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
2708 ;"JMP BR" (000110) TO GET A
2709 ;SCOPE LOOP AND HIT CONTINUE
2710
2711 012062 000442 BR IND2X ;GO TO SCOPE
2712 012064 023727 000776 000000 IND2: CME 776,04 ;STATUS SAVED ON THE STACK (Z BIT SET)
2713 012072 001405 BEQ 18 ;(RESULT OF MOV #0,-402(6))
2714
2715 012074 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2716 012076 012713 000217 MOV #217,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
2717 012102 000000 HLT ;ERROR: INCORRECT STATUS SAVED
  
```

```

2717 012104 000240          LOOP          ;REPLACE THIS INSTRUCTION BY A
2718                                ;"JMP 000" (000110) TO GET A
2719                                ;SCOPE LOOP AND HIT CONTINUE
2720 012106 022737 012054 000774 16:  CMP      01ND04,774 ;SAVED RETURN ADDRESS CORRECT?
2721 012114 001405                                BEQ      28
2722                                ;
2723 012116 005212          INC      (R2)          ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2724 012120 012713 000226          MOV      0224,(R3)    ;MOVE TO MAIL BOX (FATAL) THE ERROR
2725 012124 000000          HLT     ;ERROR! RETURN PC NOT SAVED
2726 012126 000240          LOOP    ;REPLACE THIS INSTRUCTION BY A
2727                                ;"JMP 000" (000110) TO GET A
2728                                ;SCOPE LOOP AND HIT CONTINUE
2729 012130 020627 000774          28:    CMP      00,0774
2730 012133 001405                                BEQ      36
2731                                ;
2732 012136 005212          INC      (R2)          ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2733 012140 012713 000226          MOV      0224,(R3)    ;MOVE TO MAIL BOX (FATAL) THE ERROR
2734 012144 000000          HLT     ;ERROR! INCORRECT STACK POINTER
2735 012146 000240          LOOP    ;REPLACE THIS INSTRUCTION BY A
2736                                ;"JMP 000" (000110) TO GET A
2737                                ;SCOPE LOOP AND HIT CONTINUE
2738 012150 005737 000376          38:    TST      376
2739 012153 001405                                BEQ      1ND2A
2740                                ;WAS MOVE INST. EXECUTED?
2741                                ;
2742 012156 005212          INC      (R2)          ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2743 012160 012713 000226          MOV      0224,(R3)    ;MOVE TO MAIL BOX (FATAL) THE ERROR
2744 012164 000000          HLT     ;ERROR! MOVE NOT EXECUTED
2745 012166 000240          LOOP    ;REPLACE THIS INSTRUCTION BY A
2746                                ;"JMP 000" (000110) TO GET A
2747                                ;SCOPE LOOP AND HIT CONTINUE
2748 012170 010700          IND2X: SCOPE
2749                                ;
2750                                ;*****
2751                                ;TFST 11) TEST YELLOW ZONE STACK VIOLATION, USING INDEX MODE (340,LT,R6,LT,400)
2752                                ;*****
2753 012172 005237 000304          TST11: INC      005237N
2754 012176 012700 000376          MOV      0376,06 ;SET UP STACK IN "YELLOW" AREA
2755 012202 012730 000004          MOV      01ND3,4   ;SET UP ERROR RETURN
2756 012210 005037          CLR      374
2757 012214 005037          CLR      372
2758 012220 005066 177742          CLR      -36(6)    ;FINAL ADDRESS IS 340 (YELLOW)
2759                                ;
2760 012224 005212          INC      (R2)          ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2761 012226 012713 000226          MOV      0223,(R3)    ;MOVE TO MAIL BOX (FATAL) THE ERROR
2762 012232 000000          HLT     ;FAILED TO TRAP
2763 012234 000240          LOOP    ;REPLACE THIS INSTRUCTION BY A
2764                                ;"JMP 000" (000110) TO GET A
2765                                ;SCOPE LOOP AND HIT CONTINUE
2766 012236 022737 000004 000376 1ND1:  CMP      01,374
2767 012244 001405                                BEQ      18
2768                                ;SAVED STATUS CORRECT?
2769                                ;
2770 012246 005212          INC      (R2)          ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2771 012250 012713 000226          MOV      0224,(R3)    ;MOVE TO MAIL BOX (FATAL) THE ERROR
2772 012254 000000          HLT     ;SAVED STATUS WAS INCORRECT (Z SET)
2773 012256 000240          LOOP    ;REPLACE THIS INSTRUCTION BY A

```

```

2773                                ;"JMP 000" (000110) TO GET A
2774                                ;SCOPE LOOP AND HIT CONTINUE
2775 012260 022737 012224 000372 18:  CMP      01ND10,372 ;SAVED RETURN ADDRESS CORRECT?
2776 012266 001405                                BEQ      28
2777                                ;
2778 012270 005212          INC      (R2)          ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2779 012272 012713 000226          MOV      0225,(R3)    ;MOVE TO MAIL BOX (FATAL) THE ERROR
2780 012276 000000          HLT     ;ERROR! SAVED RETURN ADDRESS INCORRECT
2781 012300 000240          LOOP    ;REPLACE THIS INSTRUCTION BY A
2782                                ;"JMP 000" (000110) TO GET A
2783                                ;SCOPE LOOP AND HIT CONTINUE
2784 012302 022706 000372          28:    CMP      0372,06
2785 012306 001405                                BEQ      36
2786                                ;STACK POINTER CORRECT
2787                                ;
2788 012310 005212          INC      (R2)          ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2789 012312 012713 000226          MOV      0226,(R3)    ;MOVE TO MAIL BOX (FATAL) THE ERROR
2790 012316 000000          HLT     ;ERROR! STACK POINTER INCORRECT
2791 012320 000240          LOOP    ;REPLACE THIS INSTRUCTION BY A
2792                                ;"JMP 000" (000110) TO GET A
2793                                ;SCOPE LOOP AND HIT CONTINUE
2794 012322 005737 000340          38:    TST      340
2795 012326 001405                                BEQ      46
2796                                ;WAS CLEAN INSTRUCTION EXECUTED?
2797                                ;
2798 012330 005212          INC      (R2)          ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2799 012332 012713 000226          MOV      0227,(R3)    ;MOVE TO MAIL BOX (FATAL) THE ERROR
2800 012336 000000          HLT     ;ERROR! CLEAN NOT EXECUTED
2801 012340 000240          LOOP    ;REPLACE THIS INSTRUCTION BY A
2802                                ;"JMP 000" (000110) TO GET A
2803                                ;SCOPE LOOP AND HIT CONTINUE
2804 012342 010700          48:    SCOPE
2805                                ;
2806                                ;*****
2807                                ;TFST 112 TEST RED ZONE STACK VIOLATION & ABORT OF THE VIOLATING INSTRUCTION
2808                                ;*****
2809 012344 005237 000304          TST112: INC      005237N
2810 012350 012700 001000          MOV      11000,06
2811 012354 012737 012436 000004          MOV      01ND4,4
2812 012362 012737 177777 000344          MOV      0-1,344   ;PRE SET SAVED STATUS LOCATION
2813 012370 012737 177777 000400          MOV      0-1,0
2814 012378 012737 177777 000002          MOV      0-1,2
2815 012386 005037 000776          CLR      776
2816 012414 005037 177776          CLR      CC
2817 012422 012737 177777 000336          MOV      0-1,336   ;PRE SET 336
2818 012426 005066 177336          CLR      -442(6)   ;CLEAR "RED" LOCATION (336)
2819                                ;
2820 012426 005212          INC      (R2)          ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2821 012430 012713 000226          MOV      0230,(R3)    ;MOVE TO MAIL BOX (FATAL) THE ERROR
2822 012434 000000          HLT     ;ERROR! FAILED TO TRAP
2823 012436 005706          IND4:  TST      06
2824 012440 001405                                BEQ      18
2825                                ;STACK POINTER CORRECT?
2826                                ;
2827 012442 005212          INC      (R2)          ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
2828 012444 012713 000231          MOV      0231,(R3)    ;MOVE TO MAIL BOX (FATAL) THE ERROR
2829 012450 000000          HLT     ;ERROR! INCORRECT STACK POINTER
2830 012452 000240          LOOP    ;REPLACE THIS INSTRUCTION BY A

```



```

2941 013012 010700 IND12: SCOPE
2942
2943 ;*****
2944 ;TEST 117 TEST RFD ZONE VIOLATION USING EMT INSTRUCTION
2945 ;*****
2946 013014 005217 000304 TST117: INC 005TESTM
2947 013020 012706 000300 MOV 0340,06 ;SET UP STACK POINTER IN "RED" AREA
2948 013024 012737 013112 000030 MOV 0RED,EMTVEC
2949 013032 005037 000032 CLR EMTVEC+2
2950 013036 012737 013116 000004 MOV 0RFR1,4
2951 013044 005037 000006 CLR 0
2952 013050 005037 000000 CLR 0
2953 013054 012737 136336 000036 MOV 0136336,336 ;PRESRT "RED" LOCATION
2954 013062 012737 177777 000002 MOV 0-1,2 ;N HIT SET
2955 013070 104000 EMT
2956 013072
2957
2958 013072 005217 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2959 013074 012713 000202 MOV 0242,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2960 013100 000000 RED10: HLT
2961 013102 000240 LOOP
2962 ;REPLACE THIS INSTRUCTION BY A
2963 ;"JMP 0R0" (000110) TO GET A
2964 ;SCOPE LOOP AND HIT CONTINUE
2965 013104 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2966 013106 012713 000243 MOV 0243,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2967 013112 000000 RED1: HLT ;DID NOT ABOPT EMT
2968 013114 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
2969 ;"JMP 0R0" (000110) TO GET A
2970 ;SCOPE LOOP AND HIT CONTINUE
2971 013116 022737 000000 000002 RED1A: CMP 00,2
2972 013124 001405 BEQ 10 ;WAS (NEW) STATUS SAVED?
2973
2974 013126 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2975 013130 012713 000244 MOV 0244,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2976 013134 000000 HLT ;CORRECT (NEW) PSW WAS NOT SAVED
2977 ;ON A RFD ZONE TRAP (CAUSED BY EMT)
2978 ;UPON EXECUTING EMT RFD ZONE OCCURED
2979 ;THE 11/00 SAVES THE "NEW" PSW WHICH
2980 ;WOULD HAVE BEEN LOADED BY THE EMT.
2981 ;11/00 SAVES THE PSW THAT WAS CURRENT AT THE
2982 ;TIME EMT WAS EXECUTED.)
2983 ;REPLACE THIS INSTRUCTION BY A
2984 ;"JMP 0R0" (000110) TO GET A
2985 ;SCOPE LOOP AND HIT CONTINUE
2986 013140 022737 013072 000000 10: CMP 0REDIC,0
2987 013146 001405 BEQ 26 ;WAS RETURN PC SAVED
2988 ;(EMTVEC)
2989 013150 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2990 013152 012713 000245 MOV 0245,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2991 013156 000000 HLT
2992 013160 000240 LOOP
2993 ;REPLACE THIS INSTRUCTION BY A
2994 ;"JMP 0R0" (000110) TO GET A
2995 ;SCOPE LOOP AND HIT CONTINUE
2996 013162 005706 20: TST 00
2997 013164 001405 BEQ 38 ;STACK POINTER=0?
    
```

```

2997
2998 013166 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2999 013170 012713 000246 MOV 0246,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3000 013174 000000 HLT
3001 013176 000240 LOOP
3002 ;REPLACE THIS INSTRUCTION BY A
3003 ;"JMP 0R0" (000110) TO GET A
3004 ;SCOPE LOOP AND HIT CONTINUE
3005 013200 022737 136336 000036 30: CMP 0136336,336
3006 013206 001405 BEQ 45 ;WAS "RED" LOCATION UNDISTURBED?
3007
3008 013210 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3009 013212 012713 000247 MOV 0247,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3010 013220 000000 HLT
3011 013222 000240 LOOP
3012 ;REPLACE THIS INSTRUCTION BY A
3013 ;"JMP 0R0" (000110) TO GET A
3014 ;SCOPE LOOP AND HIT CONTINUE
3015 013222 012737 000032 000030 40: MOV 0EMTVEC+2,EMTVEC
3016 013230 010700 SCOPE
3017
3018 ;*****
3019 ;TEST 12A TEST RFD ZONE VIOLATION USING "TRAP" INSTRUCTION
3020 ;*****
3021 013232 005217 000304 TST120: INC 005TESTM
3022 013236 012706 000200 MOV 0200,06 ;SET UP STACK IN "RED" AREA
3023 013242 012737 176176 000176 000034 MOV 0176176,176 ;PRF SET "RFD" LOCATION
3024 013250 012737 013326 000034 MOV 0RED2,TRAPVEC
3025 013256 005037 000036 CLR TRAPVEC+2
3026 013262 005037 000006 CLR 0
3027 013266 005037 000000 CLR 0
3028 013272 012737 013332 000004 MOV 0RED2,4
3029 013280 005037 000002 CLR 2 ;7 BIT SET
3030 013300 TRAP
3031 013306 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3032 013310 012713 000250 MOV 0250,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3033 013314 000000 RED20: HLT ;ERROR: FAILED TO TRAP
3034 013316 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
3035 ;"JMP 0R0" (000110) TO GET A
3036 ;SCOPE LOOP AND HIT CONTINUE
3037
3038 013320 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3039 013322 012713 000251 MOV 0251,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3040 013326 000000 RED2: HLT ;DID NOT ABOPT TRAP
3041 013330 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
3042 ;"JMP 0R0" (000110) TO GET A
3043 ;SCOPE LOOP AND HIT CONTINUE
3044 013332 022737 000000 000002 RED2A: CMP 00,2
3045 013340 001405 BEQ 16 ;WAS (NEW) STATUS SAVED?
3046
3047 013342 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3048 013344 012713 000252 MOV 0252,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3049 013350 000000 HLT ;CORRECT STATUS WORD WAS NOT SAVED
3050 ;UPON A RED ZONE VIOLATION CAUSED BY
3051 ;"TRAP". 11/00 SAVES THE NEW PSW
3052 ;WHICH WOULD HAVE BEEN LOADED BY "TRAP"
    
```

```
3053  
3054  
3055 013352 000210 LOOP ;11/40 SAVES THE "OLD" PSW THAT WAS CURRENT  
;AT THE TIME "TRAP" WAS EXECUTED  
;REPLACE THIS INSTRUCTION BY A  
;"JMP BR" (000110) TO GET A  
;SCOPE LOOP AND HIT CONTINUE  
3056 013354 022737 013306 000000 10: CMP #RED2C,0  
3059 013362 001405 ;(TRAPVEC)  
3061 013364 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
3062 013366 012713 000257 MOV #0254,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
3063 013372 000000 HLT ;REPLACE THIS INSTRUCTION BY A  
3064 013374 000210 LOOP ;"JMP BR" (000110) TO GET A  
;SCOPE LOOP AND HIT CONTINUE  
3065  
3066  
3067 013376 005706 20: TST #0  
3069 013400 001405 AFU #3  
3070  
3071 013402 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
3072 013404 012713 000254 MOV #0254,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
3073 013410 000000 HLT ;REPLACE THIS INSTRUCTION BY A  
3074 013412 000210 LOOP ;"JMP BR" (000110) TO GET A  
;SCOPE LOOP AND HIT CONTINUE  
3075  
3076 013414 022737 100176 000176 30: CMP #170176,176  
3077 013422 001405 AFU #4 ;WAS "RED" LOCATION LEFT UNDISTURBED?  
3078  
3079 013424 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
3080 013426 012713 000255 MOV #0255,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
3081 013432 000000 HLT ;PREFIX "RED" LOCATION WAS CHANGED  
3082 013434 000210 LOOP ;REPLACE THIS INSTRUCTION BY A  
3083 ;"JMP BR" (000110) TO GET A  
3084 ;SCOPE LOOP AND HIT CONTINUE  
3085 013436 012737 000016 000034 40: MOV #TRAPVEC+2,TRAPVEC  
3086 013444 010700 SCOPE  
3087  
3088 ;*****  
3089 ;*TEST 121 TEST RED ZONE VIOLATION USING "IOT" INSTRUCTION  
3090 ;*****  
3091 013446 005237 000104 TST121: INC #06TESTM  
3092 013452 012706 000100 MOV #1000,06  
3093 013456 012717 013534 000020 MOV #RED3,IOTVEC  
3094 013464 005037 000022 CLR IOTVEC+2  
3095 013472 005037 000000 CLR 0  
3096 013474 012737 013540 000004 MOV #RED3L,4  
3097 013502 005037 000000 CLR 0  
3098 013506 005037 000002 CFB 2  
3099 013512 000001 INT  
3100 013514 RED3C:  
3101  
3102 013514 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
3103 013516 012713 000256 MOV #0256,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
3104 013522 000000 HLT ;DID NOT OVERFLOW  
3105 013524 000210 LOOP ;REPLACE THIS INSTRUCTION BY A  
3106 ;"JMP BR" (000110) TO GET A  
3107 ;SCOPE LOOP AND HIT CONTINUE  
3108
```

```
3109 013526 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
3110 013530 012713 000257 MOV #0257,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
3111 013534 000000 HIT #0 ;DID NOT ABORT IOT  
3112 013536 000210 LOOP ;REPLACE THIS INSTRUCTION BY A  
3113 ;"JMP BR" (000110) TO GET A  
3114 ;SCOPE LOOP AND HIT CONTINUE  
3115 013540 022737 000000 000002 000034: CMP #0,2  
3116 013546 001405 AFU #1 ;WAS STATUS SAVED?  
3117  
3118 013550 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
3119 013552 012713 000260 MOV #0260,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
3120 013556 000000 HLT ;CORRECT STATUS WORD WAS NOT SAVED  
3121 ;UPON A RED ZONE VIOLATION CAUSED BY  
3122 ;"IOT". 11/40 SAVES THE NEW PSW  
3123 ;WHICH WOULD HAVE BEEN LOADED BY "IOT"  
3124 ;11/40 SAVES THE "OLD" PSW THAT WAS CURRENT  
3125 ;AT THE TIME "TRAP" WAS EXECUTED  
3126 013560 000210 LOOP ;REPLACE THIS INSTRUCTION BY A  
3127 ;"JMP BR" (000110) TO GET A  
3128 ;SCOPE LOOP AND HIT CONTINUE  
3129 013562 022737 013514 000100 10: CMP #RED3C,0  
3130 013570 001405 AFU #1 ;WAS RETURN PC SAVED?  
3131 ;(IOTVEC)  
3132 013572 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
3133 013574 012713 000261 MOV #0261,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
3134 013600 000000 HLT ;REPLACE THIS INSTRUCTION BY A  
3135 013602 000210 LOOP ;"JMP BR" (000110) TO GET A  
3136 ;SCOPE LOOP AND HIT CONTINUE  
3137  
3138 013604 005706 20: TST #0  
3139 013606 001405 AFU #3  
3140  
3141 013610 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
3142 013612 012713 000262 MOV #0262,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
3143 013616 000000 HLT ;REPLACE THIS INSTRUCTION BY A  
3144 013620 000210 LOOP ;"JMP BR" (000110) TO GET A  
3145 ;SCOPE LOOP AND HIT CONTINUE  
3146  
3147 013622 012737 000022 000020 30: MOV #IOTVEC+2,IOTVEC  
3148 013630 010700 SCOPE  
3149  
3150 ;*****  
3151 ;*TEST 122 TEST RED ZONE VIOLATION UPON TRACE TRAP  
3152 ;*****  
3153 013632 005737 000104 TST122: INC #06TESTM  
3154 013636 005040 CLR #0  
3155 013640 012717 013716 000014 000014: MOV #RED4,TRTVEC  
3156 013646 005037 000016 000016: CLR TRTVEC+2  
3157 013652 005037 000000 CLR 0  
3158 013656 012717 013722 000004 000004: MOV #RED4L,4  
3159 013664 005037 000000 CLR 0  
3160 013670 005037 000002 CLR 2  
3161 013674 000001 BPT ;TRACE TRAP  
3162 013676 RED4C:  
3163  
3164 013676 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
```



```

3165 013700 012713 000263      MOV      #263,(R3)      ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3166 013704 000000      HLT      ;DID NOT OVERFLOW
3167 013706 000244      LOOP    ;REPLACE THIS INSTRUCTION BY A
3168                      ;"JMP BR#" (000110) TO GET A
3169                      ;SCOPE LOOP AND HIT CONTINUE
3170
3171 013710 005212      INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3172 013712 012713 000264      MOV      #264,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3173 013716 000000      HLT      ;DID NOT ABORT TRY
3174 013720 000000      LOOP    ;REPLACE THIS INSTRUCTION BY A
3175                      ;"JMP BR#" (000110) TO GET A
3176                      ;SCOPE LOOP AND HIT CONTINUE
3177 013722 022737 000000 000002 RED4L:  CMP      #0,2          ;WAS CORRECT PSM SAVED?
3178 013730 001405      BEQ     15
3179
3180 013732 005212      INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3181 013734 012713 000265      MOV      #265,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3182 013740 000000      HLT      ;CORRECT STATUS WORD WAS NOT SAVED
3183                      ;UPON A RED ZONE TRAP CAUSED BY A BPT
3184                      ;NEW PSM TAG WOULD BE LOADED BY BPT
3185                      ;SHOULD BE SAVED.
3186 013742 000244      LOOP    ;REPLACE THIS INSTRUCTION BY A
3187                      ;"JMP BR#" (000110) TO GET A
3188                      ;SCOPE LOOP AND HIT CONTINUE
3189                      ;WAS RETURN PC SAVED?
3190 013744 022737 013076 000000 18:  CMP      #RED0+C,0
3191 013752 001405      BEQ     78
3192                      ;(TMTVEC)
3193
3194 013754 005212      INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3195 013756 012713 000266      MOV      #266,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3196 013762 000000      HLT      ;REPLACE THIS INSTRUCTION BY A
3197 013764 000244      LOOP    ;"JMP BR#" (000110) TO GET A
3198                      ;SCOPE LOOP AND HIT CONTINUE
3199                      ;WAS RETURN PC SAVED?
3200                      ;(TMTVEC)
3201
3202 013766 005212      INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3203 013770 012713 000267      MOV      #267,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3204 013774 000000      HLT      ;REPLACE THIS INSTRUCTION BY A
3205 013776 000244      LOOP    ;"JMP BR#" (000110) TO GET A
3206                      ;SCOPE LOOP AND HIT CONTINUE
3207
3208 014004 012737 000016 000014 35:  MOV      #TPTVEC+2,TPTVEC
3209 014012 010700      SCOPE
3210
3211                      ;*****
3212                      ;TPTST 123 TEST TRANSITION FROM "YELLOW" TO "RED" AREAS
3213                      ;*****
3214                      ;TST123: INC      #1,ST1V
3215                      ;THE TRANSITION OCCURS AFTER THE EMT HAS PUSHED ITS RETURN ADDRESS AND STATUS.
3216 014024 012737 000344      MOV      #344,00      ;SET UP STACK TO ALLOW 2 "PUSHES"
3217 014032 005037 000032      MOV      #REDS,EMTVEC ;LOAD EMT VECTOR
3218 014036 012737 014132 000004      CLP      EMTVEC+2     ;AND STATUS
3219 014044 012737 000000      MOV      #REDS+4      ;LOAD OVERFLOW VECTOR
3220 014052 012737 000017 000002      MOV      #1,0         ;AND STATUS
  
```

```

3221 014060 005037 000000      CLM      0
3222 014064 012737 116336 000030      MOV      #136336,336  ;PHE SET "RED" LOCATION
3223 014072 012737 000004 177776      MOV      #4,CC
3224 014100 104000      EMT
3225 014102      RED5C:
3226
3227 014102 005212      INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3228 014104 012713 000270      MOV      #270,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3229 014110 000000      HLT      ;REPLACE THIS INSTRUCTION BY A
3230 014112 000240      LOOP    ;"JMP BR#" (000110) TO GET A
3231                      ;SCOPE LOOP AND HIT CONTINUE
3232
3233 014114 000240      NOP
3234
3235 014116 005212      INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3236 014120 012713 000271      MOV      #271,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3237 014124 000000      HLT      ;REPLACE THIS INSTRUCTION BY A
3238 014126 000240      LOOP    ;"JMP BR#" (000110) TO GET A
3239                      ;SCOPE LOOP AND HIT CONTINUE
3240
3241 014130 000240      NOP
3242 014132 022737 000004 000342 RED5A:  CMP      #4,342       ;WAS STATUS SAVED?
3243 014140 001105      BEQ     16
3244
3245 014142 005212      INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3246 014144 012713 000272      MOV      #272,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3247 014150 000000      HLT      ;ERROR! EMT DID NOT SAVE STATUS
3248 014152 000240      LOOP    ;REPLACE THIS INSTRUCTION BY A
3249                      ;"JMP BR#" (000110) TO GET A
3250                      ;SCOPE LOOP AND HIT CONTINUE
3251                      ;WAS RETURN PC SAVED?
3252
3253 014164 005212      INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3254 014166 012713 000273      MOV      #273,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3255 014172 000000      HLT      ;ERROR! RETURN PC NOT SAVED
3256 014174 000240      LOOP    ;REPLACE THIS INSTRUCTION BY A
3257                      ;"JMP BR#" (000110) TO GET A
3258                      ;SCOPE LOOP AND HIT CONTINUE
3259                      ;WAS "RED" LOCATION LEFT UNDISTURBED?
3260
3261 014176 022737 136336 000336 28:  CMP      #136336,336
3262 014200 001405      BEQ     38
3263
3264 014206 005212      INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3265 014210 012713 000274      MOV      #274,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3266 014214 000000      HLT      ;ERROR! "RED" AREA WAS CHANGED
3267 014216 000240      LOOP    ;REPLACE THIS INSTRUCTION BY A
3268                      ;"JMP BR#" (000110) TO GET A
3269                      ;SCOPE LOOP AND HIT CONTINUE
3270                      ;WAS CORRECT PSM SAVED?
3271
3272 014230 005212      INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3273 014232 012713 000275      MOV      #275,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3274 014236 000000      HLT      ;CORRECT PSM WAS NOT SAVED WHEN
3275                      ;A RED ZONE OCCURED WHILE TRYING
3276                      ;TO SERVICE YELLOW ZONE, AFTER THE EMT
  
```

```

3277                                     ;IS EXECUTED THE PENDING YELLOW ZONE OCCURS
3278                                     ;(AND YELLOW ZONE PSW IS LOADED), BUT TRYING
3279                                     ;TO PUSH THE STACK CAUSES RED ZONE AND
3280                                     ;THE NEW PSW (YELLOW ZONE'S) SHOULD
3281                                     ;BE SAVED ON THE STACK
3282 014240 000740      LOOP                ;REPLACE THIS INSTRUCTION BY A
3283                                     ;"JMP BR0" (000110) TO GET A
3284                                     ;SCOPE LOOP AND HIT CONTINUE
3285 014242 072737 014124 000000 48:    CMP      #REDS.W          ;WAS PWT'S VECTOR SAVED?
3286 014250 001405      RFQ              58
3287                                     ;
3288 014252 005212      INC      (R2)                ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3289 014254 012713 000276      MOV      #276,(P3)          ;MOVE TO MAIL BOX (FATAL) THE ERROR
3290 014260 000000      HLT                    ;ERROR! OVERFLOW'S VECTOR (4) NOT SAVED
3291 014262 000240      LOOP                ;REPLACE THIS INSTRUCTION BY A
3292                                     ;"JMP BR0" (000110) TO GET A
3293                                     ;SCOPE LOOP AND HIT CONTINUE
3294 014264 005706      58:    TST      06                ;
3295 014266 001405      BEQ      05                ;
3296                                     ;
3297 014270 005212      INC      (R2)                ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3298 014272 012713 000277      MOV      #277,(P3)          ;MOVE TO MAIL BOX (FATAL) THE ERROR
3299 014276 000000      HLT                    ;
3300 014300 000240      LOOP                ;REPLACE THIS INSTRUCTION BY A
3301                                     ;"JMP BR0" (000110) TO GET A
3302                                     ;SCOPE LOOP AND HIT CONTINUE
3303 014302      68:
3304                                     ;
3305 014302 012737 000032 000030      MOV      #ENTVEC+2,ENTVEC
3306 014310 005037 000032      CLN      ENTVEC+2
3307 014314 010700      SCOPE
3308                                     ;
3309                                     ;*****
3310                                     ;TEST 124 TEST TRANSITION FROM "YELLOW" TO "RED" ZONES
3311                                     ;*****
3312 014316 005237 000304      TST124: INC      004TESTN
3313                                     ;THE TRANSITION OCCURS AFTER THE JSR HAS "PUSHED" ITS OLD R5.
3314                                     ;
3315 014322 012706 000342      MOV      #342,R6 ;SET UP STACK ON THE "HAIRY EDGE"
3316 014326 012737 014424 000004      MOV      #REDA0,R4          ;LOAD OVERFLOW VECTOR
3317 014334 012737 000357 000006      MOV      #357,R6          ;AND OVERFLOW STATUS
3318 014342 005037 000000      CLN      0
3319 014346 005037 000002      CLR      2
3320 014352 005037 000340      CLR      340
3321 014356 012705 000007      MOV      #7,R5            ;PRE SET R5
3322 014362 012737 000017 177776      MOV      #17,CC           ;PRE SET THE STATUS WORD
3323 014370 004537 014416      JSR      5,REDA0         ;SHOULD CAUSE OVERFLOW TRAP
3324 014374      RED6C:
3325                                     ;
3326 014374 005212      JMC      (R2)            ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3327 014376 012713 000308      MOV      #300,(R3)        ;MOVE TO MAIL BOX (FATAL) THE ERROR
3328 014402 000000      HLT                    ;ERROR! DID NOT OVERFLOW TRAP
3329 014404 000240      LOOP                ;REPLACE THIS INSTRUCTION BY A
3330                                     ;"JMP BR0" (000110) TO GET A
3331                                     ;SCOPE LOOP AND HIT CONTINUE
3332 014406 000240      NOP
  
```

```

3333                                     ;
3334 014410 005212      INC      (R2)            ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3335 014412 012713 000301      MOV      #301,(R3)        ;MOVE TO MAIL BOX (FATAL) THE ERROR
3336 014416 000000      HLT                    ;ERROR! DID NOT TRAP BEFORE JSR
3337 014420 000240      LOOP                ;REPLACE THIS INSTRUCTION BY A
3338                                     ;"JMP BR0" (000110) TO GET A
3339                                     ;SCOPE LOOP AND HIT CONTINUE
3340 014422 000240      NOP
3341 014424 022737 000357 177776  RED6A:  CMP      #357,CC
3342 014432 001405      BEQ      05
3343                                     ;
3344 014434 005212      INC      (R2)            ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3345 014436 012713 000302      MOV      #302,(R3)        ;MOVE TO MAIL BOX (FATAL) THE ERROR
3346 014442 000000      HLT                    ;WRONG PSW ON RED ZONE TRAP
3347 014444 000240      LOOP                ;REPLACE THIS INSTRUCTION BY A
3348                                     ;"JMP BR0" (000110) TO GET A
3349                                     ;SCOPE LOOP AND HIT CONTINUE
3350 014446 022737 000407 000340  68:    CMP      #7,340
3351 014454 001405      BEQ      15
3352                                     ;
3353 014456 005212      INC      (R2)            ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3354 014460 012713 000303      MOV      #303,(R3)        ;MOVE TO MAIL BOX (FATAL) THE ERROR
3355 014464 000000      HLT                    ;ERROR! R5 NOT SAVED ON THE STACK
3356 014466 000240      LOOP                ;REPLACE THIS INSTRUCTION BY A
3357                                     ;"JMP BR0" (000110) TO GET A
3358                                     ;SCOPE LOOP AND HIT CONTINUE
3359 014470 022705 014374      18:    CMP      #RED6C,R5
3360 014474 001405      RFQ      28
3361                                     ;
3362 014476 005212      INC      (R2)            ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3363 014500 012713 000304      MOV      #304,(R3)        ;MOVE TO MAIL BOX (FATAL) THE ERROR
3364 014504 000000      HLT                    ;ERROR! R5 DID NOT GET RETURN ADRS.
3365 014506 000240      LOOP                ;REPLACE THIS INSTRUCTION BY A
3366                                     ;"JMP BR0" (000110) TO GET A
3367                                     ;SCOPE LOOP AND HIT CONTINUE
3368 014510 022737 000357 000002  28:    CMP      #357,2
3369 014516 001405      BEQ      38
3370                                     ;
3371 014520 005212      INC      (R2)            ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3372 014522 012713 000305      MOV      #305,(R3)        ;MOVE TO MAIL BOX (FATAL) THE ERROR
3373 014526 000000      HLT                    ;CORRECT PSW WAS NOT SAVED WHEN
3374                                     ;A RED ZONE OCCURED WHILE TRYING
3375                                     ;TO SERVICE YELLOW ZONE. AFTER THE JSR
3376                                     ;IS EXECUTED THE PENDING YELLOW ZONE OCCURS
3377                                     ;(AND YELLOW ZONE PSW IS LOADED), BUT TRYING
3378                                     ;TO PUSH THE STACK CAUSES RED ZONE AND
3379                                     ;THE NEW PSW (YELLOW ZONE'S) SHOULD
3380                                     ;BE SAVED ON THE STACK
3381 014530 000240      LOOP                ;REPLACE THIS INSTRUCTION BY A
3382                                     ;"JMP BR0" (000110) TO GET A
3383                                     ;SCOPE LOOP AND HIT CONTINUE
3384 014532 022737 014416 000000  38:    CMP      #RED6,0
3385 014540 001405      BEQ      48
3386                                     ;
3387 014542 005212      INC      (R2)            ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3388 014544 012713 000306      MOV      #306,(R3)        ;MOVE TO MAIL BOX (FATAL) THE ERROR
  
```

```

3389 014550 000000 HLT
3390 014552 000240 LOOP
3391
3392
3393 014554 022737 136336 000336 46: CMP 0136336,336
3394 014562 001405 BEQ 58
3395
3396 014564 005712 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3397 014566 012713 000307 MOV 0307,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3398 014572 000000 HLT ;ERROR! "RED" LOCATION WAS CHANGED
3399 014574 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
3400 ;"JMP 000" (000110) TO GET A
3401 ;SCOPE LOOP AND HIT CONTINUE
3402 014576 010700 58: SCOPE
3403
3404
3405
3406 ;*****
3407 014600 005237 000304 TST125: INC 005237 ;TEST 125 TEST TRANSITION FROM "YELLOW" TO "RED" ZONE
3408 ;THIS TEST IS THE SAME AS ABOVE EXCEPT THAT THE TRANSITION IS AFTER THE
3409 ;OVERFLOW PUSHES ONE WORD.
3410 014604 012706 000344 MOV 0344,R6
3411 014610 012737 014676 000004 MOV 014676,R4 ;LOAD OVERFLOW VECTOR
3412 014616 015037 000006 CLR 6
3413 014622 005037 000342 CLF 342
3414 014626 012737 136336 000336 MOV 0136336,336 ;PRE SET "RED" LOCATION
3415 014634 012705 000007 MOV 07,R5 ;PRE SET R5
3416 014640 012737 000357 177776 MOV 0357,CC ;PRE SET STATUS
3417 014646 004537 014672 JSR 5,RED7 ;CAUSE OVERFLOW
3418 014652
3419
3420 014652 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3421 014654 012713 000310 MOV 0310,(R1) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3422 014660 000000 HLT ;ERROR! DID NOT OVERFLOW TRAP
3423 014662 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
3424 ;"JMP 000" (000110) TO GET A
3425 ;SCOPE LOOP AND HIT CONTINUE
3426
3427 014664 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3428 014666 012713 000311 MOV 0311,(R1) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3429 014672 000000 HLT ;ERROR! DID NOT TRAP BEFORE JSR
3430 014674 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
3431 ;"JMP 000" (000110) TO GET A
3432 ;SCOPE LOOP AND HIT CONTINUE
3433 014676 022737 000007 000342 RED7A: CMP 07,342
3434 014704 001405 BEQ 16 ;WAS R5 SAVED
3435
3436 014706 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3437 014710 012713 000312 MOV 0312,(R1) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3438 014714 000000 HLT ;ERROR! R5 NOT SAVED ON THE STACK
3439 014716 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
3440 ;"JMP 000" (000110) TO GET A
3441 ;SCOPE LOOP AND HIT CONTINUE
3442 014720 022735 014652 18: CMP 0RED7C,R5
3443 014724 001405 BEQ 28 ;DOES R5 CONTAIN RETURN PC?
3444

```

```

3445 014726 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3446 014730 012713 000313 MOV 0313,(R1) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3447 014734 000000 HLT ;ERROR! JSR DID NOT LOAD R5
3448 014736 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
3449 ;"JMP 000" (000110) TO GET A
3450 ;SCOPE LOOP AND HIT CONTINUE
3451 014740 022737 000357 000340 28: CMP 0357,340
3452 014746 001405 BEQ 38 ;WAS STATUS SAVED?
3453
3454 014750 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3455 014752 012713 000314 MOV 0314,(R1) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3456 014756 000000 HLT ;ERROR! STATUS NOT SAVED
3457 014760 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
3458 ;"JMP 000" (000110) TO GET A
3459 ;SCOPE LOOP AND HIT CONTINUE
3460 014762 022737 000000 000002 38: CMP 00,2
3461 014770 001405 BEQ 48 ;WAS STATUS SAVED?
3462
3463 014772 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3464 014774 012713 000315 MOV 0315,(R1) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3465 015000 000000 HLT ;CORRECT PSW WAS NOT SAVED WHEN
3466 ;A RED ZONE OCCURRED WHILE TRYING
3467 ;TO SERVICE YELLOW ZONE. AFTER THE JSR
3468 ;IS EXECUTED THE PENDING YELLOW ZONE OCCURS
3469 ;(AND YELLOW ZONE PSW IS LOADED), BUT TRYING
3470 ;TO PUSH THE STACK CAUSES RED ZONE AND
3471 ;THE NEW PSW (YELLOW ZONE'S) SHOULD
3472 ;BE SAVED ON THE STACK
3473 015002 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
3474 ;"JMP 000" (000110) TO GET A
3475 ;SCOPE LOOP AND HIT CONTINUE
3476 015004 022737 014672 000000 48: CMP 0RED7,0
3477 015012 001405 BEQ 58 ;RETURN PC SAVED?
3478
3479 015014 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3480 015016 012713 000316 MOV 0316,(R1) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3481 015022 000000 HLT ;ERROR! RETURN PC NOT SAVED
3482 015024 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
3483 ;"JMP 000" (000110) TO GET A
3484 ;SCOPE LOOP AND HIT CONTINUE
3485 015026 022737 136336 000336 58: CMP 0136336,336
3486 015034 001405 BEQ 68 ;WAS "RED" LOCATION UNDISTURBED?
3487
3488 015036 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3489 015040 012713 000317 MOV 0317,(R1) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3490 015044 000000 HLT ;ERROR! "RED" LOCATION WAS CHANGED
3491 015046 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
3492 ;"JMP 000" (000110) TO GET A
3493 ;SCOPE LOOP AND HIT CONTINUE
3494 015050 010700 68: SCOPE
3495
3496
3497
3498 ;*****
3499 015052 005237 000304 TST126: INC 005237 ;TEST 126 TEST THAT A TTY INTERRUPT CAUSES AN OVERFLOW TRAP
3500 015056 012737 000340 177776 MOV 0340,STATUS ;LOCK OUT INTERRUPT

```

```

3541 015064 012700 000400      MOV     #000,000      ;SET UP STACK TO OVERFLOW
3542 015074 012737 015142 000001      MOV     #TDEC7,4     ;SET UP OVERFLOW TRAP
3543 015076 012737 015136 000004      MOV     #TDEC0,4     ;SET UP INTERRUPT VECTOR
3544 015104 012737 000100 177564      MOV     #100,TTCSP   ;SET INTERRUPT ENABLE
3545 015112 005037 177770      CLR     STATUS       ;ALLOW INTERRUPT TO OCCUR
3546
3547 015116 015212              INC     (R2)         ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3548 015120 012713 000320      MOV     #320,(R3)   ;MOVE TO MAIL BOX (SFATAL) THE ERROR
3549 015124 000000      HLT
3550 015126 000211      LOOP
;REPLACE THIS INSTRUCTION BY A
;"JMP #00" (000110) TO GET A
;SCOPE LOOP AND HIT CONTINUE
3551
3552
3553
3554 015130 005212              INC     (R2)         ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3555 015132 012713 000321      MOV     #321,(R3)   ;MOVE TO MAIL BOX (SFATAL) THE ERROR
3556 015136 000000      TDEC0: HLT
3557 015140 000240      LOOP
;REPLACE THIS INSTRUCTION BY A
;"JMP #00" (000110) TO GET A
;SCOPE LOOP AND HIT CONTINUE
3558
3559 015142 005037 177564      TDEC7: CLW     TTC5H
3560 015146 010700      SCOPE

```

```

3532
3533
3534
3535
3536 015150 005237 000304      TST127: INC     #18TESTN
3537 015154 012706 021774      MOV     #BUFF,000   ;SET UP STACK POINTER
3538 015160 012737 015214 000004      MOV     #R7TR1,4    ;RETURN FROM TRAP
3539 015166 012707 000001      MOV     #1,000      ;PC EQUALS ONE
3540
3541 015172 005212              INC     (R2)         ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3542 015174 012713 000322      MOV     #322,(R3)   ;MOVE TO MAIL BOX (SFATAL) THE ERROR
3543 015200 000000      HLT
3544 015202 000240      LOOP
;REPLACE THIS INSTRUCTION BY A
;"JMP #00" (000110) TO GET A
;SCOPE LOOP AND HIT CONTINUE
3545
3546 015204 005212              INC     (R2)         ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3547 015206 012713 000323      MOV     #323,(R3)   ;MOVE TO MAIL BOX (SFATAL) THE ERROR
3548 015212 000000      HLT
3549 015214 022737 000003 021770 07TR1: CMP     #3,BUFF-4
3550 015222 001405      R00      IS
;CORRECT PC WAS NOT SAVED ON STACK
3551
3552 015224 005212              INC     (R2)         ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3553 015226 012713 000324      MOV     #324,(R3)   ;MOVE TO MAIL BOX (SFATAL) THE ERROR
3554 015232 000000      HLT
3555 015234 000240      LOOP
;REPLACE THIS INSTRUCTION BY A
;"JMP #00" (000110) TO GET A
;SCOPE LOOP AND HIT CONTINUE
3556
3557 015236
3558
3559 015236 010700      16:      SCOPE
3560
3561
3562
3563
3564
3565
3566
3567
3568
3569 015240 005237 000304      TST130: INC     #18TESTN
3570 015244 012706 021774      MOV     #BUFF,000   ;STACK POINTER
3571 015250 012737 015302 000004      MOV     #R7TR2,4    ;PC BECOMES ODD
3572 015256 005207      R7TR2: INC     #7
3573
3574 015260 005212              INC     (R2)         ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3575 015262 012713 000325      MOV     #325,(R3)   ;MOVE TO MAIL BOX (SFATAL) THE ERROR
3576 015266 000000      HLT
3577 015270 000240      LOOP
;REPLACE THIS INSTRUCTION BY A
;"JMP #00" (000110) TO GET A
;SCOPE LOOP AND HIT CONTINUE
3578
3579 015272 005212              INC     (R2)         ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3580 015274 012713 000326      MOV     #326,(R3)   ;MOVE TO MAIL BOX (SFATAL) THE ERROR
3581 015300 000000      HLT
3582 015302 022737 015263 021770 07TR2: CMP     #R7TR2+3,BUFF-4
3583 015310 001405      000      IS
;CORRECT PC NOT ON STACK
3584
3585 015312 005212              INC     (R2)         ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3586 015314 012713 000327      MOV     #327,(R3)   ;MOVE TO MAIL BOX (SFATAL) THE ERROR
3587 015320 000000      HLT

```

T130 TEST THAT A TRAP OCCURS WHEN THE PC IS INCREMENTED TO AN ODD VALUE

```
3578 015322 000240          LOUP          ;REPLACE THIS INSTRUCTION BY A
3579                                     ;"JMP 000" (000110) TO GET A
3580                                     ;SCOPE LOOP AND HIT CONTINUE
3581 015324 010700          18: SCOPE
3582                                     ;*****
3583                                     ;*TEST 131 TEST THAT A DECREMENT OF PC TO AN ODD VALUE RESULTS IN A TRAP
3584                                     ;*****
3585
3586 015326 005237 000304          TST131: INC 005TESTN
3587 015327 012736 021774          MOV 0RUFF,06
3588 015328 012737 015300 000000          MOV 0RTR1,04
3589 015329 005207          DEC 07          ;MAKE PC ODD
3590 015326
3591
3592 015326 005212          INC (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3593 015327 012713 000330          MOV 0330,(R1)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3594 015328 000000          HLT              ;SHOULD TRAP
3595 015326 000240          LOOP            ;REPLACE THIS INSTRUCTION BY A
3596                                     ;"JMP 000" (000110) TO GET A
3597                                     ;SCOPE LOOP AND HIT CONTINUE
3598 015300 002737 015347 021774 07TR3: CMP 07TR3+1,0RUFF-4 ;CHECK VALUE OF PC ON STACK
3599 015306 001345          BEQ 15
3600
3601 015327 005212          INC (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3602 015328 012713 000331          MOV 0331,(R1)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3603 015329 000000          HLT              ;WRONG VALUE ON STACK
3604 015326 000240          LOOP            ;REPLACE THIS INSTRUCTION BY A
3605                                     ;"JMP 000" (000110) TO GET A
3606                                     ;SCOPE LOOP AND HIT CONTINUE
3607
3608 015402 010700          18: SCOPE
3609                                     ;*****
3610                                     ;*TEST 132 TEST THAT "SEC. PUL PC" RESULTS IN TRAP
3611                                     ;*****
3612
3613 015404 005237 000304          TST132: INC 005TESTN
3614 015405 012736 021774          MOV 0RUFF,06
3615 015406 012737 015450 000000          MOV 0RTR4,04
3616 015407 000761          SEC
3617 015408 006107          PDL 07          ;CARRY EQUALS A 1
3618 015406          TR44:          ;PC BECOMES ODD
3619
3620 015406 005212          INC (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3621 015407 012713 000332          MOV 0332,(R1)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3622 015408 000000          HLT              ;WRONG VALUE ON STACK
3623 015406 000240          LOOP            ;REPLACE THIS INSTRUCTION BY A
3624                                     ;"JMP 000" (000110) TO GET A
3625                                     ;SCOPE LOOP AND HIT CONTINUE
3626
3627 015406 005212          INC (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3628 015407 012713 000333          MOV 0333,(R1)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3629 015408 000000          HLT              ;WRONG VALUE ON STACK
3630 015406 002737 033057 021774 07TR4: CMP 07TR4+TR4+3,0RUFF-4 ;CHECK FOR VALUE ON STACK
3631 015407 001405          BEQ 15
3632
3633 015406 005212          INC (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
```

T132 TEST THAT "SEC. HOL PC" RESULTS IN TRAP

```
3634 015402 012713 000330          MOV 0330,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3635 015403 000000          HLT              ;WRONG VALUE ON STACK
3636 015402 000240          LOOP            ;REPLACE THIS INSTRUCTION BY A
3637                                     ;"JMP 000" (000110) TO GET A
3638                                     ;SCOPE LOOP AND HIT CONTINUE
3639 015402 012737 000000 000000 18: MOV 06,0         ;RESET UP & HALT FOR TRAP
3640
3641 015500 010700          SCOPE
3642                                     ;*****
3643                                     ;*TEST 133 TEST THAT A PENDING TTY INTERRUPT OCCURS BEFORE A "TRAP" TRAP
3644                                     ;*****
3645
3646 015502 005237 000304          TST133: INC 005TESTN
3647 015503 012706 021774          MOV 0RUFF,06
3648 015512 012737 000340 177776          MOV 0340,STATUS ;HIGHEST PRIORITY LEVEL
3649 015520 012737 000100 177564          MOV 0100,TTCSR  ;INTERUPT FOR TTY PUNCH/PRINTER
3650 015526 012737 015504 000034          MOV 0TR1,04     ;TRAP VECTOR
3651 015534 012737 015570 000064          MOV 0TR2,04     ;TTY VECTOR
3652 015532 012737 000340 000036          MOV 0340,06     ;IF TRAP TRAPS, MOVE 340 TO PRIORITY
3653 015550 005037 177776          CLR STATUS      ;SHOULD TRAP AT END OF CLR INST
3654 015554 100000          TRAP            ;TTY INTERRUPT SHOULD OVERRIDE TRAP
3655
3656 015506 005212          INC (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3657 015507 012713 000335          MOV 0335,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3658 015508 000000          HLT              ;TRAP OCCUR FIRST
3659 015506 000240          LOOP            ;REPLACE THIS INSTRUCTION BY A
3660                                     ;"JMP 000" (000110) TO GET A
3661                                     ;SCOPE LOOP AND HIT CONTINUE
3662 015570 005037 177564          TP2:  CLR TTCSR
3663 015574 005037 000036          CLR 36
3664 015600 010700          SCOPE
3665
3666                                     ;*****
3667                                     ;*TEST 134 TEST THAT A PENDING TTY INTERRUPT OCCURS BETWEEN "TRAP" & AN "IOT"
3668                                     ;*****
3669
3670 015602 005237 000304          TST134: INC 005TESTN
3671 015603 012706 021774          MOV 0RUFF,06
3672 015612 012737 000340 177776          MOV 0340,STATUS ;HIGHEST PRIORITY LEVEL
3673 015620 012737 000100 177564          MOV 0100,TTCSR  ;INTERUPT FOR TTY PUNCH/PRINTER
3674 015626 012737 015604 000034          MOV 0TR1,04     ;TRAP VECTOR
3675 015634 005037 000036          CLR 36          ;TTY VECTOR
3676 015640 012737 015700 000064          MOV 0TR4,04     ;IF TRAP TRAPS, MOVE 340 TO PRIORITY
3677 015646 012737 015674 000020          MOV 0TR5,20     ;SHOULD TRAP AT END OF CLR INST
3678 015662 012737 000340 000022          TRAP 0340,02   ;TTY INTERRUPT SHOULD OVERRIDE TRAP
3679 015664 000000          IOT
3680
3681 015606 005212          INC (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3682 015607 012713 000336          MOV 0336,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3683 015608 000000          HLT              ;NO INTERRUPT BETWEEN TRAPS
3684 015606 000240          LOOP            ;REPLACE THIS INSTRUCTION BY A
3685                                     ;"JMP 000" (000110) TO GET A
3686                                     ;SCOPE LOOP AND HIT CONTINUE
3687
3687 015700 005037 177564          TP4:  CLR TTCSR
3688 015704 005037 000022          CLR 22
3689 015710 010700          SCOPE          ;CLR IOT PRIORITY
```

T135 TEST THAT 'T' BIT TRAP OCCURS BEFORE AN INTERRUPT IS ACKNOWLEDGED

```
3690 ;*****  
3691 ;TEST 135 TEST THAT 'T' BIT TRAP OCCURS BEFORE AN INTERRUPT IS ACKNOWLEDGED  
3692 ;*****  
3693 015712 005237 000304 T8T135: INC 000TESTM  
3694 015716 012737 000340 177776 MOV 0340,CC ;SET PRIORITY =7  
3695 015724 012737 000100 177564 MOV 0100,TTCSR ;ENABLE INTERRUPT ON TTY PRINTER  
3696 015732 012737 015770 000014 MOV INT,TRITVEC ;LOAD 'T' BIT VECTOR  
3697 015740 005037 000016 CLA TRITVEC+2  
3698 015744 012737 016002 000064 MOV 0INT,64 ;LOAD TTY INT. VECTOR  
3699 015752 012706 021774 MOV 0BUFF,06 ;SET STACK POINTER  
3700 015756 012746 000020 MOV 020,+(6) ;PUSH 'T'BIT ON THE STACK  
3701 015762 012746 015770 MOV 0,+(6) ;PUSH PC ON THE STACK  
3702 015766 000002 RTI ;SET 'T' BIT  
3703 015770  
3704  
3705 015770 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
3706 015772 012713 000337 MOV 0337,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
3707 015776 000000 HLT ;ERROR! TTY INTERRUPTED  
3708 016000 000240 LOOP ;REPLACE THIS INSTRUCTION BY A  
3709 ;"JMP 000" (000110) TO GET A  
3710 ;SCOPE LOOP AND HIT CONTINUE  
3711 016002 005037 177564 INT2: CLR TTCSR  
3712 016006 027237 000020 021772 CMP 020,BUFF-2  
3713 016014 001405 BEQ 15 ;DISABLE TTY INTERRUPT  
3714  
3715 016016 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
3716 016020 012713 000340 MOV 0340,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
3717 016024 000000 HLT  
3718 016026 000240 LOOP ;REPLACE THIS INSTRUCTION BY A  
3719 ;"JMP 000" (000110) TO GET A  
3720 ;SCOPE LOOP AND HIT CONTINUE  
3721 016030 027237 015770 021770 16: CMP 0INT,BUFF-4  
3722 016036 001405 BEQ 25  
3723  
3724 016040 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
3725 016042 012713 000341 MOV 0341,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
3726 016046 000000 HLT  
3727 016050 000240 LOOP ;REPLACE THIS INSTRUCTION BY A  
3728 ;"JMP 000" (000110) TO GET A  
3729 ;SCOPE LOOP AND HIT CONTINUE  
3730 016052 027237 000000 021766 26: CMP 00,BUFF-6  
3731 016060 001405 BEQ 30  
3732  
3733 016062 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
3734 016064 012713 000342 MOV 0342,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
3735 016070 000000 HLT  
3736 016072 000240 LOOP ;REPLACE THIS INSTRUCTION BY A  
3737 ;"JMP 000" (000110) TO GET A  
3738 ;SCOPE LOOP AND HIT CONTINUE  
3739 016074 027237 015770 021764 36: CMP 0INT,BUFF-10  
3740 016102 001405 BEQ INT1  
3741  
3742 016104 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
3743 016106 012713 000343 MOV 0343,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
3744 016112 000000 HLT  
3745 016114 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
```

T136 TEST THAT 'T' BIT TRAP OCCURS BEFORE AN INTERRUPT IS ACKNOWLEDGED

```
3746 ;"JMP 000" (000110) TO GET A  
3747 ;SCOPE LOOP AND HIT CONTINUE  
3748 016116 010700 INT1: SCOPE  
3749  
3750  
3751 ;*****  
3752 ;TEST 136 TEST THAT "RESET" GOES TO OUTSIDE WORLD  
3753 ;*****  
3754 016120 005237 000304 TST136: INC 000TESTM  
3755 016124 012706 021774 MOV 0BUFF,06  
3756 016130 012737 000340 177776 MOV 0340,CC ;LOCK OUT INTERRUPTS  
3757 016136 012737 016206 000064 MOV 0TR6,64 ;LOAD TELEPRINTER VECTOR  
3758 016144 012737 000100 177564 MOV 0100,TTCSR ;SET INTERRUPT ENABLE  
3759 016152 000005 RESET ;SHOULD CLEAR INTERRUPT ENABLE  
3760 016154 032737 000100 177564 BIT 0100,TTCSR ;TEST FOR CLEAR  
3761 016162 001405 BEQ 15  
3762 016164 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
3763 016166 012713 000344 MOV 0344,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
3764 016172 000000 HLT ;RESET FAILED TO CLEAR TTCSR  
3765 016174 000240 LOOP ;REPLACE THIS INSTRUCTION BY A  
3766 ;"JMP 000" (000110) TO GET A  
3767 ;SCOPE LOOP AND HIT CONTINUE  
3768 016176 000406 16: BR TR6X ;GO TO SCOPE  
3769  
3770 016200 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR  
3771 016202 012713 000345 TP6: MOV 0345,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR  
3772 016206 000000 HLT ;ERROR! TELETYPE INTERRUPTED WHEN  
3773 ;PROCESSOR WAS AT LEVEL 7  
3774 016210 000240 LOOP ;REPLACE THIS INSTRUCTION BY A  
3775 ;"JMP 000" (000110) TO GET A  
3776 ;SCOPE LOOP AND HIT CONTINUE  
3777 016212 022626 TR6X: CMP 06,+(6)+  
3778 016214 005037 177564 CLR TTCSR  
3779 016220 010700 SCOPE  
3780  
3781 ;*****  
3782 ;TEST 137 TEST THAT RESET DOES NOT HANG THE SYSTEM  
3783 ;*****  
3784 016222 005237 000304 TST137: INC 000TESTM  
3785 016226 012706 021774 MOV 0BUFF,06 ;SET STACK  
3786 016232 005037 177776 CLR STATUS ;ALLOW INTERRUPT  
3787 016236 012737 016252 000064 MOV 0RESET,64 ;TTY INTERRUPT VECTOR  
3788 016244 005237 000100 177564 BIS 0100,TTCSR ;SET INTERRUPT ENABLE  
3789 016252 000005 RESET: RESET ;IF THIS HANGS CHECK SACK  
3790 016254 012737 000066 000064 MOV 066,64 ;FOR FALSE INTERRUPT  
3791 016262 010700 SCOPE  
3792 ;*****  
3793 ;TEST 140 TEST RESET WITH TRACE ON  
3794 ;*****  
3795 016264 005237 000304 TST140: INC 000TESTM  
3796 016270 012706 021774 MOV 0BUFF,06 ;SET STACK  
3797 016274 012737 016336 000014 MOV 0RESET,TRITVEC ;SET UP TRACE VECTOR  
3798 016302 005037 000016 CLR TRITVEC+2  
3799 016306 012746 000020 MOV 020,+(6) ;PUSH 'T'BIT ON THE STACK  
3800 016312 012746 016320 MOV 0,+(6) ;PUSH PC ON THE STACK  
3801 016316 000002 RTI ;SET 'T' BIT
```

```

3802 016320 000005          RESET
3803 016322 000005          RRESET          ;SHOULD HAVE NO EFFECT
3804                                     ;NO EFFECT
3805 016324 005212          INC (R2)
3806 016326 012713 000346  MOV 0346,(R3) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3807 016332 000000          HLT          ;MOVE TO MAIL BOX (FATAL) THE ERROR
3808 016334 000240          LOOP        ;TRACE TRAP FAILED
3809                                     ;REPLACE THIS INSTRUCTION BY A
3810                                     ;"JMP 000" (000110) TO GET A
3811 016336 005037 177776    RESET2: CLR STATUS ;SCOPE LOOP AND HIT CONTINUE
3812 016342 005037 000016    CLR 16      ;CLEAR TRACK
3813 016346 010700          SCOPE      ;TRACE STATUS
3814
3815
3816
3817
3818 016350 005217 000304    TST141: INC 010700
3819 016354 000005          HRESET
3820 016356 012706 021774    MOV 0BUFF,06 ;SET UP STACK
3821 016362 012737 016406 000064  MOV 0TTY3,64 ;INTERRUPT VECTOR
3822 016370 005037 177776    CLR STATUS ;DPOP PROCESSOR PRIORITY
3823 016374 012737 000357 000066  MOV 0357,06 ;HIGH PRIORITY ON INTERRUPT
3824 016402 005137 177564    COM TTCSR ;SHOULD SET INTERRUPT ENABLE & INTERRUPT
3825
3826 016406 013727 177776    TTY3: MOV STATUS,(PC)+ ;SAVE PROCESSOR STATUS
3827 016412 000000          _WORD 0
3828 016414 022737 000357 016412  CMP 0357,.-2
3829 016422 001405          HEQ 18
3830
3831 016424 005712          INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3832 016426 012713 000347  MOV 0347,(R3) ;MOVE TO MAIL BOX (FATAL) THE ERROR
3833 016432 000000          HLT          ;INTERRUPT DID NOT POP CORRECT STATUS
3834 016434 000240          LOOP        ;REPLACE THIS INSTRUCTION BY A
3835                                     ;"JMP 000" (000110) TO GET A
3836                                     ;SCOPE LOOP AND HIT CONTINUE
3837 016436 000005          L0: HRESET ;CLR INTERRUPT ENABLE
3838 016440 010700          SCOPE
3839 016442 012706 021774    MOV 0BUFF,06 ;STACK SET UP
3840 016446 012737 016472 000064  MOV 0TTY4,64 ;INTERRUPT VECTOR
3841 016454 005037 000006  CLR 06      ;CLF NEW STATUS
3842 016460 012737 000157 177776  MOV 0157,STATUS ;PROCESSOR STATUS
3843 016466 005137 177564    COM TTCSR ;SET INTERRUPT ENABLE
3844 016472 013727 177776    TTY4: MOV STATUS,(PC)+ ;SAVE NEW STATUS
3845 016476 000000          _WORD 0
3846 016500 005737 016476    TST .-2
3847 016504 001405          HEQ 18
3848
3849 016506 005212          INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3850 016510 012713 000350  MOV 0350,(R3) ;MOVE TO MAIL BOX (FATAL) THE ERROR
3851 016514 000000          HLT          ;INTERRUPT DID NOT POP CORRECT STATUS
3852 016516 000240          LOOP        ;REPLACE THIS INSTRUCTION BY A
3853                                     ;"JMP 000" (000110) TO GET A
3854                                     ;SCOPE LOOP AND HIT CONTINUE
3855 016520 005037 177564    L6: CLR TTCSR
3856 016524 010700          SCOPE
3857

```

```

3858
3859
3860 016526 005237 000304    TST142: INC 010700
3861                                     ;THIS TEST CHECKS THAT ALL MEMORY (UP TO 20K) IS CONTIGUOUS
3862 016532 012706 021774    MOV 0BUFF,06 ;SET STACK POINTER
3863 016536 012737 016552 000004  MOV 0MEMEND,004 ;SET TIME OUT INTR VECTOR
3864 016544 005000          CLR 00      ;SET STARTING ADDRESS FOR TEST
3865 016546 005720          TST (0)+   ;BEGIN
3866 016550 000776          BR .-2     ;LOOP UNTIL TIMEOUT OCCURS
3867 016552 022626 160002  MEMEND: CMP (61+,16)+ ;RESTORE STACK POINTER
3868 016554 022700 160002  CMP 0160002,00 ;AT END OF 20K MEMORY??
3869 016560 001406          BEQ MEMEX ;YES = EXIT TEST
3870 016562 005720          TST (0)+   ;SHOULD TIME OUT HERE UP TO 20K
3871
3872 016564 005212          INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3873 016566 012713 000351  MOV 0351,(R3) ;MOVE TO MAIL BOX (FATAL) THE ERROR
3874 016572 000000          HLT          ;ERROR = FAILED TO TIME OUT!
3875 016574 000240          LOOP        ;REPLACE THIS INSTRUCTION BY A
3876                                     ;"JMP 000" (000110) TO GET A
3877                                     ;SCOPE LOOP AND HIT CONTINUE
3878 016576 012737 000006 000004  MEMEX: MOV 06,004 ;RESTORE TRAPCATCHER
3879 016604 005037 000006  CLR 006
3880 016610 010700          SCOPE

```

```
3081 ;SPECIAL CASE OF ODD, EVEN, BYTE AND REGISTER b
3082
3083
3084 ;
3085 ;*****
3086 ;TEST 143 TEST AUTO INCREMENT & DECREMENT OF R6 = WORD & BYTE
3087 ;*****
3088 TST(R3) INC R6,TESTRN
3089
3090 AIDRb1 MOV #RUFF,R6 ;SET UP THE STACK
3091 MOVR (A)+,(C)+ ;SIX SHOULD INCREMENT BY TWO
3092 M ,WORD
3093 CMF @6,#RUFF+2
3094 BCU 18
3095
3096 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3097 MOV #352,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
3098 HLT ;R6 DID NOT AUTO INCREMENT BY TWO
3099 LOOP ;REPLACE THIS INSTRUCTION BY A
3100 ;"JMP #R0" (000110) TO GET A
3101 ;SCOPE LOOP AND HIT CONTINUE
3102
3103
3104 18: SCOPE
3105
3106 MOV #RUFF,R6
3107 MOVR (A)+,(C)+ ;SHOULD DECREMENT BY TWO
3108 M ,WORD
3109 CMF @6,#776
3110 BEW 25
3111
3112 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3113 MOV #353,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
3114 HLT ;R6 DID NOT AUTO DECREMENT BY 2
3115 LOOP ;REPLACE THIS INSTRUCTION BY A
3116 ;"JMP #R0" (000110) TO GET A
3117 ;SCOPE LOOP AND HIT CONTINUE
3118
3119
3120 25: SCOPE
3121
3122 MOV #RUFF,R6
3123 MOVR (A)+,(C)+ ;DOUBLE AUTO INCREMENT OF R6
3124 CMF @6,#RUFF+4
3125 BEQ 35
3126
3127 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3128 MOV #354,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
3129 HLT ;WRONG AUTO INCREMENT OF R6
3130 LOOP ;REPLACE THIS INSTRUCTION BY A
3131 ;"JMP #R0" (000110) TO GET A
3132 ;SCOPE LOOP AND HIT CONTINUE
3133
3134
3135 35: SCOPE
3136
3137 MOV #RUFF,R6
3138 CLR @4
3139 CMFB (6)+,(4)+ ;TEST INCREMENT OF R6
3140 CMP @6,#RUFF+2
3141 BEQ 45
3142
3143 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3144 MOV #355,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
3145 HLT ;WRONG INCREMENT OF R6
3146 LOOP ;REPLACE THIS INSTRUCTION BY A
3147 ;"JMP #R0" (000110) TO GET A
3148 ;SCOPE LOOP AND HIT CONTINUE
3149
3150
3151 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3152 MOV #356,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
3153 HLT ;WRONG INCREMENT OF R6
3154 LOOP ;REPLACE THIS INSTRUCTION BY A
3155 ;"JMP #R0" (000110) TO GET A
3156 ;SCOPE LOOP AND HIT CONTINUE
3157
3158
3159 45: SCOPE
3160
3161 MOV #RUFF,R6
3162 CLR @4
3163 CMFB (6)+,(4)+ ;TEST INCREMENT OF R4
3164 CMP @6,#1
3165 BEQ 55
3166
3167 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3168 MOV #357,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
3169 HLT ;WRONG INCREMENT OF R4
3170 LOOP ;REPLACE THIS INSTRUCTION BY A
3171 ;"JMP #R0" (000110) TO GET A
3172 ;SCOPE LOOP AND HIT CONTINUE
3173
3174
3175 55: SCOPE
3176
3177 MOV #RUFF,R6
3178 CLR @4
3179 CMFB (4)+,(6)+ ;TEST DECREMENT OF R6
3180 CMP @6,#RUFF-2
3181 BEQ 75
3182
3183 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3184 MOV #360,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
3185 HLT ;WRONG INCREMENT OF R6
3186 LOOP ;REPLACE THIS INSTRUCTION BY A
3187 ;"JMP #R0" (000110) TO GET A
3188 ;SCOPE LOOP AND HIT CONTINUE
3189
3190
3191 75: SCOPE
3192
3193 MOV #RUFF,R6
3194 CLR @4
3195 CMFB (4)+,(6)+ ;TEST INCREMENT OF R4
3196 CMP @6,#1
3197 BEQ 85
3198
3199 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3200 MOV #361,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
3201 HLT ;WRONG INCREMENT OF R4
```

```
3037
3038
3039
3040 ;
3041 ;*****
3042 ;TEST 143 TEST AUTO INCREMENT & DECREMENT OF R6 = WORD & BYTE
3043 ;*****
3044 TST(R3) INC R6,TESTRN
3045
3046 AIDRb1 MOV #RUFF,R6 ;SET UP THE STACK
3047 MOVR (A)+,(C)+ ;SIX SHOULD INCREMENT BY TWO
3048 M ,WORD
3049 CMF @6,#RUFF+2
3050 BCU 18
3051
3052 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3053 MOV #352,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
3054 HLT ;R6 DID NOT AUTO INCREMENT BY TWO
3055 LOOP ;REPLACE THIS INSTRUCTION BY A
3056 ;"JMP #R0" (000110) TO GET A
3057 ;SCOPE LOOP AND HIT CONTINUE
3058
3059
3060 18: SCOPE
3061
3062 MOV #RUFF,R6
3063 MOVR (A)+,(C)+ ;SHOULD DECREMENT BY TWO
3064 M ,WORD
3065 CMF @6,#776
3066 BEW 25
3067
3068 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3069 MOV #353,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
3070 HLT ;R6 DID NOT AUTO DECREMENT BY 2
3071 LOOP ;REPLACE THIS INSTRUCTION BY A
3072 ;"JMP #R0" (000110) TO GET A
3073 ;SCOPE LOOP AND HIT CONTINUE
3074
3075
3076 25: SCOPE
3077
3078 MOV #RUFF,R6
3079 MOVR (A)+,(C)+ ;DOUBLE AUTO INCREMENT OF R6
3080 CMF @6,#RUFF+4
3081 BEQ 35
3082
3083 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3084 MOV #354,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
3085 HLT ;WRONG AUTO INCREMENT OF R6
3086 LOOP ;REPLACE THIS INSTRUCTION BY A
3087 ;"JMP #R0" (000110) TO GET A
3088 ;SCOPE LOOP AND HIT CONTINUE
3089
3090
3091 35: SCOPE
3092
3093 MOV #RUFF,R6
3094 CLR @4
3095 CMFB (6)+,(4)+ ;TEST INCREMENT OF R6
3096 CMP @6,#RUFF+2
3097 BEQ 45
3098
3099 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3100 MOV #355,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
3101 HLT ;WRONG INCREMENT OF R6
3102 LOOP ;REPLACE THIS INSTRUCTION BY A
3103 ;"JMP #R0" (000110) TO GET A
3104 ;SCOPE LOOP AND HIT CONTINUE
3105
3106
3107 45: SCOPE
3108
3109 MOV #RUFF,R6
3110 CLR @4
3111 CMFB (4)+,(6)+ ;TEST DECREMENT OF R6
3112 CMP @6,#RUFF-2
3113 BEQ 55
3114
3115 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3116 MOV #356,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
3117 HLT ;WRONG INCREMENT OF R6
3118 LOOP ;REPLACE THIS INSTRUCTION BY A
3119 ;"JMP #R0" (000110) TO GET A
3120 ;SCOPE LOOP AND HIT CONTINUE
3121
3122
3123 55: SCOPE
3124
3125 MOV #RUFF,R6
3126 CLR @4
3127 CMFB (6)+,(4)+ ;TEST INCREMENT OF R4
3128 CMP @6,#1
3129 BEQ 65
3130
3131 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3132 MOV #357,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
3133 HLT ;WRONG INCREMENT OF R4
3134 LOOP ;REPLACE THIS INSTRUCTION BY A
3135 ;"JMP #R0" (000110) TO GET A
3136 ;SCOPE LOOP AND HIT CONTINUE
3137
3138
3139 65: SCOPE
3140
3141 MOV #RUFF,R6
3142 CLR @4
3143 CMFB (4)+,(6)+ ;TEST DECREMENT OF R6
3144 CMP @6,#RUFF-2
3145 BEQ 75
3146
3147 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3148 MOV #360,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
3149 HLT ;WRONG INCREMENT OF R6
3150 LOOP ;REPLACE THIS INSTRUCTION BY A
3151 ;"JMP #R0" (000110) TO GET A
3152 ;SCOPE LOOP AND HIT CONTINUE
3153
3154
3155 75: SCOPE
3156
3157 MOV #RUFF,R6
3158 CLR @4
3159 CMFB (4)+,(6)+ ;TEST INCREMENT OF R4
3160 CMP @6,#1
3161 BEQ 85
3162
3163 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
3164 MOV #361,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
3165 HLT ;WRONG INCREMENT OF R4
```



```

3993 017130 002100 LOOP
3994
3995 ;REPLACE THIS INSTRUCTION BY A
; "JMP R00" (000110) TO GET A
; SCOPE LOOP AND HIT CONTINUE
3996 017112 010700 96: SCOPE
3997
3998 017114 012700 011000 MOV R1000,R6
3999 017116 124627 011000 CMBR -(R6),(R0)+ ;TEST DECREMENT OF R6
4000 017142 022700 000770 CMP #770,R6
4001 017146 001005 BEQ #8
4002
4003 017150 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
4004 017152 (1271) 000362 MOV #362,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
4005 017156 000000 HLT ;WONG DECREMENT OF R6
4006 017160 000241 LOOP ;REPLACE THIS INSTRUCTION BY A
; "JMP R00" (000110) TO GET A
; SCOPE LOOP AND HIT CONTINUE
4008
4009 017162 010700 98: SCOPE
4010
4011 017164 010700 SCOPE ;15 JAN 71
4012 017166 012700 021774 MOV #00FF,R6 ;INITIAL POSITION OF R6
4013 017172 012737 177777 071772 MOV #-1,00FF-2 ;HIGH BYTE WILL BE DATA
4014 017200 112737 025252 001000 MOV #25252,R1 ;INITIAL SET UP OF DESTINATION
4015 017206 114637 001001 MOVH -(R6),R1 ;HIGH BYTE OF 00FF-2, TO LOW BYTE OF R1
4016 017212 022737 177652 001000 CMB #177652,R1 ;TEST RESULT0
4017 017220 001005 BEQ #0
4018
4019 017222 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
4020 017224 012711 000363 MOV #161,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
4021 017230 000000 HLT ;MOVH -(R6), 0000, FAILED
4022 017232 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
; "JMP R00" (000110) TO GET A
; SCOPE LOOP AND HIT CONTINUE
4023
4024
4025 017234 010700 100: SCOPE
4026
4027 017236 012700 021774 MOV #00FF,R6 ;INITIAL POSITION OF R6
4028 017242 012737 177777 071772 MOV #-1,00FF-2 ;HIGH BYTE SOURCE DATA
4029 017250 012737 125252 001000 MOVH #125252,R1 ;INITIAL SET UP OF DESTINATION
4030 017256 114637 001001 MOVH -(R6),R1 ;HIGH BYTE OF 00FF-2, TO LOW BYTE OF R1
4031 017262 022737 125377 001000 CMB #125377,R1 ;TEST RESULT0
4032 017270 001005 BEQ #16
4033
4034 017272 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
4035 017274 012711 000364 MOV #164,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
4036 017300 000000 HLT ;MOVH -(R6), EVEN, FAILED
4037 017302 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
; "JMP R00" (000110) TO GET A
; SCOPE LOOP AND HIT CONTINUE
4038
4039
4040 017304 010700 110: SCOPE

```

```

4041
4042 ;TEST 144 TEST TRANSFER OF .BYTE USING R6
4043 ;*****
4044 017306 005237 000304 IST144: INC #00TESTN
4045 017312 012737 123456 001010 BXP0: MOV #123456,R5
4046 017320 012737 050505 001000 MOV #050505,R1
4047 017326 012705 001000 MOV #1,R5 ;R5=(050505)R1
4048 017332 012706 001010 MOV #5,R6 ;R6(123456)R5
4049 017336 112025 MOVH (R6),(R5)+ ;LOW .BYTE OF R6 TO R5
4050 017340 022737 050456 001000 CMP #050456,R1
4051 017346 001005 BEQ #8
4052
4053 017350 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
4054 017352 012711 000365 MOV #365,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
4055 017356 000000 HLT ;FALSE TRANSFER OF .BYTE
4056 017360 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
; "JMP R00" (000110) TO GET A
; SCOPE LOOP AND HIT CONTINUE
4057
4058
4059 017362 010700 10: SCOPE
4060
4061 017364 012737 123456 001010 MOV #123456,R5
4062 017372 012737 050505 001000 MOV #050505,R1
4063 017380 012705 001000 MOV #1,R5 ;R5(050505)R1
4064 017400 012706 001010 MOV #5,R6 ;R6(123456)R5
4065 017410 114625 MOVH (R6),(R5)+ ;LOW .BYTE OF R6 TO R5 (DECREMENT)
4066 017412 022727 001000 000456 CMP #1,0050456
4067 017420 001005 BEQ #8
4068
4069 017422 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
4070 017424 012711 000366 MOV #366,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
4071 017430 000000 HLT ;FALSE R6 .BYTE TRANSFER
4072 017432 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
; "JMP R00" (000110) TO GET A
; SCOPE LOOP AND HIT CONTINUE
4073
4074
4075 017434 010700 20: SCOPE
4076
4077 017436 012737 123456 001000 MOV #123456,R1
4078 017444 012737 050505 001010 MOV #050505,R5
4079 017452 012705 001000 MOV #1,R5 ;(123456)
4080 017462 012706 001010 MOV #5,R6 ;(050505)
4081 017462 112526 MOVH (R5),(R6)+ ;LOW OF R5 TO LOW OF R6
4082 017464 022737 050456 001010 CMP #050456,R5
4083 017472 001005 BEQ #8
4084
4085 017474 005212 INC (R2) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
4086 017476 012711 000367 MOV #367,(R3) ;MOVE TO MAIL BOX (SFATAL) THE ERROR
4087 017502 000000 HLT ;FALSE R6 .BYTE TRANSFER
4088 017504 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
; "JMP R00" (000110) TO GET A
; SCOPE LOOP AND HIT CONTINUE
4089
4090
4091 017506 010700 30: SCOPE
4092
4093 017510 012737 123456 001000 MOV #123456,R1
4094 017516 012737 050505 001010 MOV #050505,R5
4095 017524 012705 001001 MOV #1,R5 ;123456
4096 017530 012706 001010 MOV #5,R6 ;050505

```

```
4097 017534 112525      MOV# (5)+,(6)+      ;HIGH OF R5 TO LOW OF R6
4098 017534 021727 001010 050647      CMP      X5,0050647
4099 017544 001405      H#0
4100
4101 017546 005212      INC      (R2)      ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
4102 017550 012713 000370      MOV      #370,(R3) ;MOVE TO MAIL BOX (FATAL) THE ERROR
4103 017554 000000      HLT
4104 017556 000240      L/0P      ;FALSE R6 ,BYTE TRANSFER
4105      ;REPLACE THIS INSTRUCTION BY A
4106      ;"JMP #R# (000110) TO GET A
4107 017560 010700      41:      SCOPE      ;SCOPE LOOP AND HIT CONTINUE
4108
4109 017562 012737 123456 001000      MOV      #123456,X1
4110 017570 012737 050505 001010      MOV      #050505,X5
4111 017576 012700 001001      MOV      #1+1,05
4112 017602 012700 001010      MOV      #X5,06
4113 017606 112625      MOV# (R)+,(5)+      ;R5-121456--ODD ADDRESS
4114 017610 022737 042456 001000      CMP      #042456,X1 ;R6-050505--EVEN ADDRESS
4115 017616 001405      H#0
4116
4117 017620 005212      INC      (R2)      ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
4118 017622 012713 000371      MOV      #371,(R3) ;MOVE TO MAIL BOX (FATAL) THE ERROR
4119 017626 000000      HLT
4120 017630 000240      LOOP     ;FAILED LOW OF # TO HIGH OF 5
4121      ;REPLACE THIS INSTRUCTION BY A
4122      ;"JMP #R# (000110) TO GET A
4123 017632 010700      54:      SCOPE      ;SCOPE LOOP AND HIT CONTINUE
4124
4125
4126
4127
4128 017634 045237 000304      TSTI45: INC      #05237N
4129 017640 123737 001014 001035      BYTOP: CMPB     #7,K7+1
4130 017646 001405      BEQ      #6      ;SAME ,WORD LOW TO HIGH
4131
4132 017650 005212      INC      (R2)
4133 017652 012713 000372      MOV      #372,(R3) ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
4134 017656 000000      HLT
4135 017660 000240      LOOP     ;MOVE TO MAIL BOX (FATAL) THE ERROR
4136      ;SHOULD COMPARE LOW TO HIGH
4137      ;REPLACE THIS INSTRUCTION BY A
4138      ;"JMP #R * ( ) TO GET A
4139 017662 010700      14:      SCOPE      ;SCOPE LOOP AND HIT CONTINUE
4140
4141 017664 123737 001015 001014      CMP#     #7+1,K7
4142 017672 001405      BEQ      #28
4143
4144 017674 005212      INC      (R2)      ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
4145 017676 012713 000373      MOV      #373,(R3) ;MOVE TO MAIL BOX (FATAL) THE ERROR
4146 017682 000000      HLT
4147 017684 000240      LOOP     ;ODD TO ,EVEN ,BYTE FAILURE
4148      ;REPLACE THIS INSTRUCTION BY A
4149      ;"JMP #R# (000110) TO GET A
4150 017686 010700      28:      SCOPE      ;SCOPE LOOP AND HIT CONTINUE
4151
4152 017710 123737 001017 001014      CMP#     #10+1,K7
4153 017716 001405      BEQ      #38      ;SEQUENTIAL ,BYTES
4154      ;DIFFERENT ,WORDS
```

```
4153
4154 017720 005212      INC      (R2)      ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
4155 017722 012713 000374      MOV      #374,(R3) ;MOVE TO MAIL BOX (FATAL) THE ERROR
4156 017726 000000      HLT
4157 017730 000240      LOOP     ;ODD TO ,EVEN FAILED
4158      ;REPLACE THIS INSTRUCTION BY A
4159      ;"JMP #R# (000110) TO GET A
4160 017732 010700      38:      SCOPE      ;SCOPE LOOP AND HIT CONTINUE
4161
4162 017734 123737 001016 001012      CMP#     #10,#6
4163 017742 001405      BEQ      #48
4164
4165 017744 005212      INC      (R2)      ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
4166 017746 012713 000375      MOV      #375,(R3) ;MOVE TO MAIL BOX (FATAL) THE ERROR
4167 017752 000000      HLT
4168 017754 000240      LOOP     ;,EVEN TO EVEN FAILED
4169      ;REPLACE THIS INSTRUCTION BY A
4170      ;"JMP #R# (000110) TO GET A
4171 017756 010700      48:      SCOPE      ;SCOPE LOOP AND HIT CONTINUE
4172
4173 017760 123737 001015 001017      CMP#     #7+1,#10+1
4174 017766 001405      BEQ      #58
4175
4176 017770 005212      INC      (R2)      ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
4177 017772 012713 000376      MOV      #376,(R3) ;MOVE TO MAIL BOX (FATAL) THE ERROR
4178 017776 000000      HLT
4179 020000 000240      LOOP     ;ODD TO ODD FAILED
4180      ;REPLACE THIS INSTRUCTION BY A
4181      ;"JMP #R# (000110) TO GET A
4182 020002 010700      58:      SCOPE      ;SCOPE LOOP AND HIT CONTINUE
4183
4184 020004 123737 001016 001017      CMP#     #10,#10+1
4185 020012 001405      BEQ      #68
4186
4187 020014 005212      INC      (R2)      ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
4188 020016 012713 000377      MOV      #377,(R3) ;MOVE TO MAIL BOX (FATAL) THE ERROR
4189 020022 000000      HLT
4190 020024 000240      LOOP     ;LOW TO HIGH IN SAME ,WORD FAILED
4191      ;REPLACE THIS INSTRUCTION BY A
4192      ;"JMP #R# (000110) TO GET A
4193 020026 010700      68:      SCOPE      ;SCOPE LOOP AND HIT CONTINUE
4194
4195 020030 123737 001017 001016      CMP#     #10+1,#10
4196 020036 001405      BEQ      #78
4197
4198 020040 005212      INC      (R2)      ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
4199 020042 012713 000400      MOV      #400,(R3) ;MOVE TO MAIL BOX (FATAL) THE ERROR
4200 020046 000000      HLT
4201 020050 000240      LOOP     ;HIGH TO LOW IN SAME ,WORD FAILED
4202      ;REPLACE THIS INSTRUCTION BY A
4203      ;"JMP #R# (000110) TO GET A
4204 020052 010700      78:      SCOPE      ;SCOPE LOOP AND HIT CONTINUE
4205
4206 020054 123737 001016 001015      CMP#     #10,#7+1
4207 020062 001405      BEQ      #88
```

```
4209 020064 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4210 020066 012713 000401    MOV      0401,(R1)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4211 020072 000000          HLT                     ;FVEN TO ODD FAILED
4212 020074 000740          LOOP                    ;REPLACE THIS INSTRUCTION BY A
4213                                     ;"JMP 000" (000110) TO GET A
4214                                     ;SCOPE LOOP AND HIT CONTINUE
4215 020076 010700          84:   SCOPE
4216
```

```
4217
4218
4219                                     ;*****
4220                                     ;TEST 146 TEST THAT MOVB BR MOVES ONLY THE LSH OF THE REGISTER
4221                                     ;*****
4221 020100 005237 000304          IST146: INC      000TESTM
4222 020104 012737 177777 001020  RLOBYT: MOV      0-1,R11
4223 020112 012700 000125          MOV      1125,00
4224 020116 110037 001020          MOVB    00,R11          ;LOAD R0
4225 020122 023727 001020 177525  CMP      R11,0177525    ;WAS ONLY LSH MOVED?
4226 020130 001405          BEQ      15
4227
4228 020132 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4229 020134 012713 000402    MOV      0402,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4230 020140 000000          HLT                     ;ERROR! MOVB BR FAILED
4231 020142 000240          LOOP                    ;REPLACE THIS INSTRUCTION BY A
4232                                     ;"JMP 000" (000110) TO GET A
4233                                     ;SCOPE LOOP AND HIT CONTINUE
4234 020144 010700          18:   SCOPE
4235
4236 020144 012700 001020          MOV      0R11,00
4237 020152 010037 001020          MOV      00,R11
4238 020156 110020          MOVB    00,(0)+
4239 020160 022737 001021 001020  CMP      R11+1,R11+2
4240 020166 001405          BEQ      24
4241
4242 020170 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4243 020172 012713 000403    MOV      0403,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4244 020176 000000          HLT                     ;REPLACE THIS INSTRUCTION BY A
4245 020200 000240          LOOP                    ;"JMP 000" (000110) TO GET A
4246                                     ;SCOPE LOOP AND HIT CONTINUE
4247
4248 020202 010700          26:   SCOPE
4249
4250 020204 012700 001020          MOV      0R11,06
4251 020210 010037 001020          MOV      06,R11
4252 020214 110026          MOVB    06,(6)+
4253 020216 023727 001020 001022  CMP      R11,0R11+2
4254 020224 001405          BEQ      38
4255
4256 020226 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4257 020230 012713 000404    MOV      0404,(R1)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4258 020234 000000          HLT                     ;FAILED MOV 06,(6)+
4259 020236 000240          LOOP                    ;REPLACE THIS INSTRUCTION BY A
4260                                     ;"JMP 000" (000110) TO GET A
4261                                     ;SCOPE LOOP AND HIT CONTINUE
4262 020240 010700          38:   SCOPE
4263
4264 020242 012700 001020          MOV      0R11,06
4265 020246 010026          MOV      06,(6)+
4266 020250 023727 001020 001022  CMP      R11,R11+2
4267 020256 001405          BEQ      48
4268
4269 020260 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4270 020262 012713 000405    MOV      0405,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4271 020266 000000          HLT                     ;FAILED MOV 06,(6)+
4272 020270 000240          LOOP                    ;REPLACE THIS INSTRUCTION BY A
```

```

4273                                     ;"JMP 000" (000110) TO GET A
4274                                     ;SCOPE LOOP AND HIT CONTINUE
4275 020272 010700          44:   SCOPF
4276                                     ;
4277 020274 000777          SCC          ;SET STATUS
4278 020276 025017 177776   CLR          ;CLEAR STATUS
4279 020302 103005          BCC          55
4280                                     ;
4281 020304 005212          INC          (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4282 020306 012713 000400   MOV          0406,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4283 020312 000000          HLT          ;Z NOT CLEAR
4284 020314 000240          LOOP         ;REPLACE THIS INSTRUCTION BY A
4285                                     ;"JMP 000" (000110) TO GET A
4286                                     ;SCOPE LOOP AND HIT CONTINUE
4287 020316 102005          56:   AYC          60
4288                                     ;
4289 020320 005212          INC          (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4290 020322 012713 000407   MOV          0407,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4291 020326 000000          HLT          ;Z NOT CLEAR
4292 020330 000240          LOOP         ;REPLACE THIS INSTRUCTION BY A
4293                                     ;"JMP 000" (000110) TO GET A
4294                                     ;SCOPE LOOP AND HIT CONTINUE
4295 020332 001005          65:   FNE          75
4296                                     ;
4297 020334 005212          INC          (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4298 020336 012713 000410   MOV          0410,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4299 020342 000000          HLT          ;Z NOT CLEAR
4300 020344 000240          LOOP         ;REPLACE THIS INSTRUCTION BY A
4301                                     ;"JMP 000" (000110) TO GET A
4302                                     ;SCOPE LOOP AND HIT CONTINUE
4303 020346 100005          70:   NPL          85
4304                                     ;
4305 020350 005212          INC          (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4306 020352 012713 000411   MOV          0411,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4307 020356 000000          HLT          ;Z NOT CLEAR
4308 020360 000240          LOOP         ;REPLACE THIS INSTRUCTION BY A
4309                                     ;"JMP 000" (000110) TO GET A
4310                                     ;SCOPE LOOP AND HIT CONTINUE
4311 020362 010700          85:   SCOPE
4312                                     ;
4313 020364 000257          CCC          ;CLEAR CONDITION CODES
4314 020366 052737 000017 177776   FIS          ;SET STATUS TO ONES
4315 020374 103405          RCS          90
4316                                     ;
4317                                     ;
4318 020376 005212          INC          (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4319 020400 012713 000412   MOV          0412,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4320 020404 000000          HLT          ;Z NOT SET
4321 020406 000240          LOOP         ;REPLACE THIS INSTRUCTION BY A
4322                                     ;"JMP 000" (000110) TO GET A
4323                                     ;SCOPE LOOP AND HIT CONTINUE
4324 020410 102405          90:   BVS          105
4325                                     ;
4326 020412 005212          INC          (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4327 020414 012713 000413   MOV          0413,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4328 020420 000000          HLT          ;Z NOT SET
  
```

```

4329 020422 000240          LOOP         ;REPLACE THIS INSTRUCTION BY A
4330                                     ;"JMP 000" (000110) TO GET A
4331                                     ;SCOPE LOOP AND HIT CONTINUE
4332 020424 001405          100:  NEQ          110
4333                                     ;
4334 020426 005212          INC          (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4335 020430 012713 000414   MOV          0414,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4336 020434 000000          HLT          ;Z NOT SET
4337 020436 000240          LOOP         ;REPLACE THIS INSTRUCTION BY A
4338                                     ;"JMP 000" (000110) TO GET A
4339                                     ;SCOPE LOOP AND HIT CONTINUE
4340 020440 100405          110:  RMI          125
4341                                     ;
4342 020442 005212          INC          (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4343 020444 012713 000415   MOV          0415,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4344 020450 000000          HLT          ;Z NOT SET
4345 020452 000240          LOOP         ;REPLACE THIS INSTRUCTION BY A
4346                                     ;"JMP 000" (000110) TO GET A
4347                                     ;SCOPE LOOP AND HIT CONTINUE
4348 020454 010700          120:  SCOPE
4349                                     ;
4350                                     ;
4351                                     ;
4352 020456 012706 021774   MOV          0BUFF,06     ;TEST THAT (06) = 4 TO 06
4353 020462 005037 021776   CLR          0BUFF+2     ;SET UP BUFFER POINTER
4354 020466 012737 020470 021774   MOV          0,+10,0UFF   ;SET UP NEW STATUS
4355 020474 000000          RTI          ;SET UP RETURN
4356 020476 020627 022000   CMP          06,0UFF+4    ;RETURN NEXT INSTRUCTION
4357 020502 001405          REQ          130        ;06 SHOULD BE PLUS 4
4358                                     ;
4359 020504 005212          INC          (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4360 020506 012713 000416   MOV          0416,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4361 020512 000000          HLT          ;06 NOT INCREMENTED
4362 020514 000240          LOOP         ;REPLACE THIS INSTRUCTION BY A
4363                                     ;"JMP 000" (000110) TO GET A
4364                                     ;SCOPE LOOP AND HIT CONTINUE
4365                                     ;
4366 020516 010700          130:  SCOPE
4367 020520 012737 000357 021776   MOV          0357,0UFF+2  ;TEST ((06+2)) TO STATUS
4368 020526 012706 021774   MOV          0UFF,06     ;RETURN STATUS ALL ONES
4369 020532 012737 020544 021774   MOV          0,+12,0UFF   ;SET UP BUFFER POINTER
4370 020540 000257          CCC          ;SET UP RETURN POINTER
4371 020542 000000          RTI          ;CLEAR CONDITION CODES
4372                                     ;RETURN
4373 020544 001405          0EU          140
4374                                     ;
4375 020546 005212          INC          (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4376 020550 012713 000417   MOV          0417,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4377 020554 000000          HLT          ;Z NOT SET
4378 020556 000240          LOOP         ;REPLACE THIS INSTRUCTION BY A
4379                                     ;"JMP 000" (000110) TO GET A
4380                                     ;SCOPE LOOP AND HIT CONTINUE
4381 020560 102405          140:  BVS          150
4382                                     ;
4383 020562 005212          INC          (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4384 020564 012713 000420   MOV          0420,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
  
```

```

4385 020574 070000      HLT
4386 020572 000240      LOOP
4387
4388
4389 020574 100405      158:  RMI      165
4390
4391 020576 005212      INC (R2)
4392 020600 012713      MOV 0421,(R3)
4393 020600 000000      HLT
4394 020606 000240      LOOP
4395
4396
4397 020610 101005      148:  RCM      175
4398
4399 020612 005212      INC (R2)
4400 020614 012713      MOV 0422,(R3)
4401 020620 000000      HLT
4402 020622 000240      LOOP
4403
4404
4405
4406 020624 010700      178:  SCOPE
4407 020626 005037      CLR 000FF,00A
4408 020632 012700      MOV 0,+12,00FF
4409 020636 012717      MOV 020650,071774
4410 020644 000000      SCC
4411 020646 000000      KIT
4412 020650 001005      UNE 105
4413
4414 020652 005212      INC (R2)
4415 020654 012713      MOV 0423,(R3)
4416 020660 000000      HLT
4417 020662 000240      LOOP
4418
4419
4420 020664 102005      188:  RVC      195
4421
4422 020666 005212      INC (R2)
4423 020670 012713      MOV 0424,(R3)
4424 020674 000000      HLT
4425 020676 000240      LOOP
4426
4427
4428 020700 100005      198:  RPL      205
4429
4430 020702 005212      INC (R2)
4431 020704 012713      MOV 0425,(R3)
4432 020710 000000      HLT
4433 020712 000240      LOOP
4434
4435
4436 020714 103005      208:  RCC      215
4437
4438 020716 005212      INC (R2)
4439 020720 012713      MOV 0426,(R3)
4440 020724 000000      HLT
  
```

```

4441 020726 000240      LOOP
4442
4443
4444 020730 010700      218:  SCOPE
4445 020732
  
```



```

4550 ;JMP RRR* (00010) TO GET A
4555 ;SCOPE LOOP AND HIT CONTINUE
4560 #21314 000410 BP SINSI
4561 #21316 022700 000001 CMP #1,00
4562 #21322 001405 RFD SINSI
4563 ;HAS INST. AT SINSI EXECUTED?
4564 #21324 005212 INC (R2) ;SET MESSAGE TYPE (&MSGTY) TO FATAL ERROR
4565 #21326 012713 000417 MOV #1437,(P3) ;MOVE TO MAIL BOX (&FATAL) THE ERROR
4566 #21332 000000 HLT ;ERROR! INST AT SINSI WAS NOT EXECUTED
4567 #21334 000740 LOOP ;REPLACE THIS INSTRUCTION BY A
4568 ;JMP RRR* (00010) TO GET A
4569 ;SCOPE LOOP AND HIT CONTINUE
4570 #21336 010700 SINSI: SCOPE
4571
4572 #21340 000137 021420 JMP ENDOT ;END OF PROGRAM, GOT TO
4573 ;END OF TEST LINKAGE
4574
4575 ;XFC= 076XXX EXCEPT 0766XX
4576 #21344 000007 TABLE: ?
4577 ;MED=076600
4578 #21346 000077 77
4579 #21350 000210 210
4580 #21352 000227 227
4581 #21354 007000 7000
4582 #21356 007777 7777
4583 #21360 075040 75040
4584 #21362 076577 76577
4585
4586 #21364 076601 76601
4587 #21366 076677 76677
4588
4589 #21370 106400 106400
4590 #21372 106477 106477
4591
4592 #21374 106700 106700
4593 #21376 107777 107777
4594 #21400 021400 FINISH:
4595 #21402 000000 INST: HLT ;END FLAG
4596 #21404 000000 HLT ;WILL CONTAIN RESERVED INST
4597 #21406 000000 HLT ;SHOULD TRAP TO LOC 10
4598 #21410 000000 HLT ;LOC 10 SHOULD SEND YOU TO
4599 #21412 000000 HLT ;RET
4600 #21414 000000 TEMP: R
4601 #21420 021420 .,+.2
4602
4603 ;END OF PASS LINKAGE
4604
4605
4606 #21420 005237 000306 ENDOT: INC #0,PASS
4607 #21424 105237 021546 INCR PASSPT ;SHOULD PRINT THIS PASS?
4608 #21430 001027 000000 BNE ACT ;NO
4609 #21432 132737 000000 000321 BTR #40,&FNVM ;WILL APT ALLOW PRINTING?
4610 #21440 001023 000000 BNE ACT ;NO
4611 #21442 023727 000042 #21520 CMP #42,&ENDAD
4612 #21450 001417 000000 BFD ACT
4613 #21452 012700 021550 MOV #MSG,R0 ;GET MSG ADDR.
  
```

```

4614 #21456 105737 177564 WAIT: TSTB #0,TPS ;TTY READY
4615 #21462 100375 000000 BPL WAIT ;NO WAIT
4616 #21464 112037 177564 MOVR (R1)+,#0,TPB ;PRINT CHARACTER
4617 #21470 001372 000000 BNE WAIT ;NEXT IF NOT DONE.
4618 #21472 105737 177564 WAIT1: TSTR #0,TPS
4619 #21476 100375 000000 BPL WAIT1
4620 #21500 000005 RFSET
4621 #21502 012737 177761 021546 MOV #177761,PASSPT ;DO IT ABOUT 15 DECIMAL TIMES
4622 #21510 013700 000042 ACT: MOV #42,PM ;CHECK ACT
4623 #21514 001405 000000 BEQ GOAGIN ;KEEP GOING
4624 #21516 000005 RESET
4625 #21520 004713 000000 BENDAD: JSR PC,(R0) ;ACT WORKS
4626 #21522 000240 NOP
4627 #21524 000240 NOP
4628 #21526 000240 NOP
4629 #21530 012737 000012 000010 COAGIN: MOV #12,10
4630 #21536 000037 000012 CLR 12
4631 #21542 000137 001110 JMP BEGIN1 ;DO NEXT PASS
4632 #21546 177777 PASSPT: -1
4633 #21550 005015 047105 020104 MSG: .ASCII2 <15><12>;END OF PASS/
4634 #21556 043117 020000 040520
4635 #21564 051523 000
4636 #21570 021570 .EVFN
4637
4638 ;POWER FAIL, POWER UP ROUTINE
4639
4640 #21570 012737 021600 000024 PWRDWN: MOV #PWRUP,24
4641 #21576 000000 HLT
4642
4643 #21600 012737 021570 000024 PWRUP: MOV #PWRDWN,24
4644 #21606 012706 021774 000000 MOV #BUFF,SP
4645 #21612 132737 000000 000321 BTR #40,&FNVM ;WILL APT ALLOW PRINTING?
4646 #21620 001011 000000 BNE PFRFS ;NO
4647 #21622 012700 021654 MOV #MSGPWF,R0 ;GET MSG ADDR.
4648 #21626 105737 177564 PWAIT: TSTR #0,TPS ;TTY READY
4649 #21632 100375 000000 BPL Pwait ;NO WAIT
4650 #21634 112037 177566 MOVN (R0)+,#0,TPB ;PRINT CHARACTER
4651 #21640 001372 000000 BNE PWAIT ;NEXT IF NOT DONE.
4652 #21642 105737 177564 PWAIT1: TSTB #0,TPS
4653 #21646 100375 000000 BPL PWAIT1
4654 #21650 000137 001102 PFRFS: JMP BFCIN
4655 #21654 005015 047520 042527 MSCPWF: .ASCII2 <15><12>.POWER FAILED!
4656 #21662 020122 040506 046111
4657 #21670 042105 000041
4658
4659 #21774 000000 .,+.100
4660 #21774 000000 BUFF: 0
4661 #21774 000001 .FND
  
```

ARASE	# 000000	310			
ACD#1	# 000000	310			
ACD#2	# 000000	310			
ACFUOP	# 000000	310			
ACT	F71510	4600	4610	4612	46220
ADD#0	# 000000	310			
ADD#1	# 000000	310			
ADD#10	# 000000	310			
ADD#11	# 000000	310			
ADD#12	# 000000	310			
ADD#13	# 000000	310			
ADD#14	# 000000	310			
ADD#15	# 000000	310			
ADD#2	# 000000	310			
ADD#3	# 000000	310			
ADD#4	# 000000	310			
ADD#5	# 000000	310			
ADD#6	# 000000	310			
ADD#7	# 000000	310			
ADD#8	# 000000	310			
ADD#9	# 000000	310			
ADVECT	# 000000	310			
ADVEY	# 000000	310	374		
AFNY	# 000000	310			
AFNYM	# 000000	310	324		
AFATAL	# 000000	310	330		
ATDPA	# 010616	30000	321		
AMADR1	# 000000	310			
AMADR2	# 000000	310			
AMADR3	# 000000	310			
AMADR4	# 000000	310			
AMADR5	# 000000	310			
AMADR6	# 000000	310			
AMADR7	# 000000	310			
AMADR8	# 000000	310			
AMADR9	# 000000	310			
AMADR10	# 000000	310			
AMADR11	# 000000	310			
AMADR12	# 000000	310			
AMADR13	# 000000	310			
AMADR14	# 000000	310			
AMADR15	# 000000	310			
AMADR16	# 000000	310			
AMADR17	# 000000	310			
AMADR18	# 000000	310			
AMADR19	# 000000	310			
AMADR20	# 000000	310			
AMADR21	# 000000	310			
AMADR22	# 000000	310			
AMADR23	# 000000	310			
AMADR24	# 000000	310			
AMADR25	# 000000	310			
AMADR26	# 000000	310			
AMADR27	# 000000	310			
AMADR28	# 000000	310			
AMADR29	# 000000	310			
AMADR30	# 000000	310			
AMADR31	# 000000	310			
AMADR32	# 000000	310			
AMADR33	# 000000	310			
AMADR34	# 000000	310			
AMADR35	# 000000	310			
AMADR36	# 000000	310			
AMADR37	# 000000	310			
AMADR38	# 000000	310			
AMADR39	# 000000	310			
AMADR40	# 000000	310			
AMADR41	# 000000	310			
AMADR42	# 000000	310			
AMADR43	# 000000	310			
AMADR44	# 000000	310			
AMADR45	# 000000	310			
AMADR46	# 000000	310			
AMADR47	# 000000	310			
AMADR48	# 000000	310			
AMADR49	# 000000	310			
AMADR50	# 000000	310			
AMADR51	# 000000	310			
AMADR52	# 000000	310			
AMADR53	# 000000	310			
AMADR54	# 000000	310			
AMADR55	# 000000	310			
AMADR56	# 000000	310			
AMADR57	# 000000	310			
AMADR58	# 000000	310			
AMADR59	# 000000	310			
AMADR60	# 000000	310			
AMADR61	# 000000	310			
AMADR62	# 000000	310			
AMADR63	# 000000	310			
AMADR64	# 000000	310			
AMADR65	# 000000	310			
AMADR66	# 000000	310			
AMADR67	# 000000	310			
AMADR68	# 000000	310			
AMADR69	# 000000	310			
AMADR70	# 000000	310			
AMADR71	# 000000	310			
AMADR72	# 000000	310			
AMADR73	# 000000	310			
AMADR74	# 000000	310			
AMADR75	# 000000	310			
AMADR76	# 000000	310			
AMADR77	# 000000	310			
AMADR78	# 000000	310			
AMADR79	# 000000	310			
AMADR80	# 000000	310			
AMADR81	# 000000	310			
AMADR82	# 000000	310			
AMADR83	# 000000	310			
AMADR84	# 000000	310			
AMADR85	# 000000	310			
AMADR86	# 000000	310			
AMADR87	# 000000	310			
AMADR88	# 000000	310			
AMADR89	# 000000	310			
AMADR90	# 000000	310			
AMADR91	# 000000	310			
AMADR92	# 000000	310			
AMADR93	# 000000	310			
AMADR94	# 000000	310			
AMADR95	# 000000	310			
AMADR96	# 000000	310			
AMADR97	# 000000	310			
AMADR98	# 000000	310			
AMADR99	# 000000	310			
AMADR100	# 000000	310			
AMADR101	# 000000	310			
AMADR102	# 000000	310			
AMADR103	# 000000	310			
AMADR104	# 000000	310			
AMADR105	# 000000	310			
AMADR106	# 000000	310			
AMADR107	# 000000	310			
AMADR108	# 000000	310			
AMADR109	# 000000	310			
AMADR110	# 000000	310			
AMADR111	# 000000	310			
AMADR112	# 000000	310			
AMADR113	# 000000	310			
AMADR114	# 000000	310			
AMADR115	# 000000	310			
AMADR116	# 000000	310			
AMADR117	# 000000	310			
AMADR118	# 000000	310			
AMADR119	# 000000	310			
AMADR120	# 000000	310			
AMADR121	# 000000	310			
AMADR122	# 000000	310			
AMADR123	# 000000	310			
AMADR124	# 000000	310			
AMADR125	# 000000	310			
AMADR126	# 000000	310			
AMADR127	# 000000	310			
AMADR128	# 000000	310			
AMADR129	# 000000	310			
AMADR130	# 000000	310			
AMADR131	# 000000	310			
AMADR132	# 000000	310			
AMADR133	# 000000	310			
AMADR134	# 000000	310			
AMADR135	# 000000	310			
AMADR136	# 000000	310			
AMADR137	# 000000	310			
AMADR138	# 000000	310			
AMADR139	# 000000	310			
AMADR140	# 000000	310			
AMADR141	# 000000	310			
AMADR142	# 000000	310			
AMADR143	# 000000	310			
AMADR144	# 000000	310			
AMADR145	# 000000	310			
AMADR146	# 000000	310			
AMADR147	# 000000	310			
AMADR148	# 000000	310			
AMADR149	# 000000	310			
AMADR150	# 000000	310			
AMADR151	# 000000	310			
AMADR152	# 000000	310			
AMADR153	# 000000	310			
AMADR154	# 000000	310			
AMADR155	# 000000	310			
AMADR156	# 000000	310			
AMADR157	# 000000	310			
AMADR158	# 000000	310			
AMADR159	# 000000	310			
AMADR160	# 000000	310			
AMADR161	# 000000	310			
AMADR162	# 000000	310			
AMADR163	# 000000	310			
AMADR164	# 000000	310			
AMADR165	# 000000	310			
AMADR166	# 000000	310			
AMADR167	# 000000	310			
AMADR168	# 000000	310			
AMADR169	# 000000	310			
AMADR170	# 000000	310			
AMADR171	# 000000	310			
AMADR172	# 000000	310			
AMADR173	# 000000	310			
AMADR174	# 000000	310			
AMADR175	# 000000	310			
AMADR176	# 000000	310			
AMADR177	# 000000	310			
AMADR178	# 000000	310			
AMADR179	# 000000	310			
AMADR180	# 000000	310			
AMADR181	# 000000	310			
AMADR182	# 000000	310			
AMADR183	# 000000	310			
AMADR184	# 000000	310			
AMADR185	# 000000	310			
AMADR186	# 000000	310			
AMADR187	# 000000	310			
AMADR188	# 000000	310			
AMADR189	# 000000	310			
AMADR190	# 000000	310			
AMADR191	# 000000	310			
AMADR192	# 000000	310			
AMADR193	# 000000	310			
AMADR194	# 000000	310			
AMADR195	# 000000	310			
AMADR196	# 000000	310			
AMADR197	# 000000	310			
AMADR198	# 000000	310			
AMADR199	# 000000	310			
AMADR200	# 000000	310			
AMADR201	# 000000	310			
AMADR202	# 000000	310			
AMADR203	# 000000	310			
AMADR204	# 000000	310			
AMADR205	# 000000	310			
AMADR206	# 000000	310			
AMADR207	# 000000	310			
AMADR208	# 000000	310			
AMADR209	# 000000	310			
AMADR210	# 000000	310			
AMADR211	# 000000	310			
AMADR212	# 000000	310			
AMADR213	# 000000	310			
AMADR214	# 000000	310			
AMADR215	# 000000	310			
AMADR216	# 000000	310			
AMADR217	# 000000	310			
AMADR218	# 000000	310			
AMADR219	# 000000	310			
AMADR220	# 000000	310			
AMADR221	# 000000	310			
AMADR222	# 000000	310			
AMADR223	# 000000	310			
AMADR224	# 000000	310			
AMADR225	# 000000	310			
AMADR226	# 000000	310			
AMADR227	# 000000	310			
AMADR228	# 000000	310			
AMADR229	# 000000	310			
AMADR230	# 000000	310			
AMADR231	# 000000	310			
AMADR232	# 000000	310			
AMADR233	# 000000	310			
AMADR234	# 000000	310			
AMADR235	# 000000	310			
AMADR236	# 000000	310			







RETC4	#03654	1053	1055*										
RETC5	#04312	1190	1192*										
RETD	#01420	400	401*										
RETD1	#02052	626	629*										
RFTD2	#02566	782	785*										
RETD3	#03220	917	920*										
RFTD4	#03724	1072	1075*										
RETD5	#04362	1209	1212*										
RETF	#01464	505	508*										
RFTF1	#02116	642	645*										
RETF2	#02632	798	801*										
RETF3	#03264	933	936*										
RETF4	#03770	1080	1083*										
RFTF5	#04426	1225	1228*										
RFTF6	#01544	527	511*										
RFTF7	#02176	663	667*										
RFTF8	#02712	819	823*										
RFTF9	#03344	954	958*										
RFTF10	#04050	1109	1113*										
RFTF11	#04506	1246	1250*										
RFTG	#01820	546	550*										
RETC1	#02262	680	684*										
RETC2	#02766	837	841*										
RETC3	#03414	971	975*										
RFTG4	#04124	1127	1131*										
RFTG5	#04562	1264	1268*										
RETH5	#04652	1287	1291*										
RFTJ	#04670	1301	1305*										
RFTK	#04732	1320	1324*										
RETL	#05002	1339	1343*										
RFTM	#05446	1355	1359*										
RETN	#05126	1376	1380*										
RFTO	#05202	1394	1398*										
RFTP	#05274	1418	1422*										
RFTQ	#05320	1436	1440*										
RFTP	#05364	1455	1459*										
RFTS	#05436	1474	1478*										
RFTT	#05504	1489	1493*										
RFTU	#05566	1509	1513*										
RFTV	#05640	1526	1530*										
RFTW	#01930	4460	4473*										
RFTX	#02052	4474	4487*										
RFTY	#21072	4483	4496*										
RFTZA	#21112	4458	4497*										
RLOBYT	#20104	4222*											
RLOBYI	#20712	4445*											
RFTP1	#15214	3520*	3544*										
RFTP2	#15302	3550*	3572*										
RFTP2A	#15260	3560*	3572*										
RFTP3	#15360	3580*	3598*										
RFTP3A	#15346	3590*	3598*										
RFTRA	#15450	3615*	3630*										
SCOPE = #10700	270*	427	441	463	482	502	520	543	565	582	601	620	639
	656	677	696	721	737	756	775	795	812	834	856	872	891
	910	930	947	968	986	1010	1027	1046	1065	1085	1102	1124	1146
	1164	1183	1202	1222	1239	1261	1280	1297	1313	1332	1352	1369	1391

1410	1420	1440	1467	1486	1502	1523	1540	1558	1571	1589	1606	1619		
1637	1676	1696	1720	1748	1787	1836	1855	1872	1880	1907	1921	1956		
1970	1989	2019	2040	2061	2082	2092	2113	2150	2179	2208	2237	2266		
2306	2556	2676	2693	2747	2802	2870	2888	2905	2923	2941	3014	3086		
3148	3200	3307	3402	3494	3521	3551	3581	3608	3641	3664	3689	3740		
3779	3791	3813	3838	3856	3880	3900	3914	3927	3941	3955	3969	3982		
3996	4009	4031	4025	4048	4059	4075	4091	4107	4123	4138	4149	4160		
4171	4182	4193	4204	4215	4234	4248	4262	4275	4311	4340	4366	4406		
4444	4497	4570												
SINSD	#21176	4506	4510*											
SINSDD	#21160	4512	4514*											
SINSE	#21222	4507	4532*											
SINSE	#21204	4500	4523*											
SINSE	#21246	4532	4542*											
SINSH	#21276	4542	4552*											
SINSI	#21336	4522	4531	4541	4560	4567	4570*							
SINSI	#21316	4543	4561*											
SINSI	#21120	4505*												
STATUS = 177776	2094	3500*	3505*	3640*	3653*	3671*	3786*	3811*	3822*	3826	3842*	3844	4270*	
	4314*													
SYKFR	#11316	2547*												
TABLE	#21344	4454	4576*											
TBITVE	#00014	276*	3696*	3697*	3797*	3798*	4506*	4543*						
TDEC1	#07034	1843	1890*											
TDEC1A	#07040	1846	1855*											
TDEC2	#07076	1862	1872*											
TDEC2A	#07134	1879	1880*											
TDEC3	#07172	1896	1902*											
TDEC4	#07224	1910	1916*											
TDEC5	#07304	1930	1941*											
TDEC5A	#07276	1936*												
TDEC5B	#07336	1940	1955	1956*										
TDEC6	#07176	1898	1907*											
TDEC6A	#07230	1912	1921*											
TDEC7	#15142	3502	3520*											
TDEC8	#15136	3503	3516*											
TFRP	#21414	1640	1647	1648*	1649*	1684*	2561*	2567*	2570*	2614	2665	2879*	2896*	2913
	2931	4600*												
TPB = 177566	285*	420*	4616*	4650*										
TPS = 177564	284	418	423	4614	4618	4648	4652	670*	663*	684*	680*	681*	719*	720*
TRAPVE = #00034	281*	572*	573*	589*	600*	620*	642*	663*	684*	680*	681*	719*	720*	
	2333*	2334*	3022*	3023*	3085*									
TRCSR = 177560	283*													
TRT = #00003	272*													
TRTA	#11410	2567	2568*	2570*	2581									
TRTA	#11414	25710	2631											
TRTM	#11416	2565	2572*											
TRTC	#11532	2563	2600	2613*										
TRTD	#11566	2616	2622*											
TRTE	#11702	2617	2640	2662*										
TRTEX	#11732	2590	2609	2640	2659	2666	2674*							
TRTF	#11710	2571	2665*											
TRTVEC = #00014	270*	1706*	1735*	1754*	1795*	1834*	1835*	2337*	2338*	2563*	2564*	2674*		
	3155*	3156*	3650*											
TR1	#15564	3650*												
TR2	#15570	3651	3662*											

TR3	015004	3673	3679*
TR4	015700	3675	3677*
TR1A	015426	3610*	361*
TR5	015674	3676	3641*
TR6	016206	3756	3772*
TR61	016214	3760	3770*
TST1	001226	431*	
TST14	001762	640*	
TST100	010030	2120*	
TST101	010106	2150*	
TST102	010164	2104*	
TST103	010202	2213*	
TST104	010320	2242*	
TST105	010376	2273*	
TST106	010506	2311*	
TST107	011332	2560*	
TST11	002026	624*	
TST110	011750	2601*	
TST111	012172	2752*	
TST112	012304	207*	
TST113	012504	2075*	
TST114	012634	2093*	
TST115	012676	2010*	
TST116	012744	2020*	
TST117	013014	2016*	
TST12	002144	641*	
TST120	013232	3019*	
TST121	013446	3001*	
TST122	013632	3151*	
TST123	014014	3213*	
TST124	014316	3312*	
TST125	014600	3407*	
TST126	015052	3490*	
TST127	015150	3520*	
TST13	002304	701*	
TST130	015240	3556*	
TST131	015326	3506*	
TST132	015404	3613*	
TST133	015502	3666*	
TST134	015602	3669*	
TST135	015712	3693*	
TST136	016120	3753*	
TST137	016222	3744*	
TST14	002374	725*	
TST140	016264	3795*	
TST141	016350	3810*	
TST142	016526	3860*	
TST143	016612	3890*	
TST144	017306	4044*	
TST145	017634	4120*	
TST146	020100	4271*	
TST147	020732	4450*	
TST15	002434	742*	
TST150	021114	4502*	
TST16	002476	761*	
TST17	002542	706*	

TST2	001266	440*	
TST20	002600	817*	
TST21	003026	860*	
TST22	003066	877*	
TST23	003130	896*	
TST24	003174	915*	
TST25	003312	952*	
TST26	003442	991*	
TST27	003532	1015*	
TST3	001334	460*	
TST30	003572	1032*	
TST31	003634	1051*	
TST32	003700	1070*	
TST33	004016	1127*	
TST34	004170	1152*	
TST35	004230	1169*	
TST36	004272	1180*	
TST37	004336	1207*	
TST4	001374	480*	
TST40	004454	1244*	
TST41	004614	1285*	
TST42	004712	1310*	
TST43	004756	1337*	
TST44	005074	1374*	
TST45	005234	1416*	
TST46	005276	1430*	
TST47	005342	1453*	
TST5	001512	525*	
TST50	005430	1472*	
TST51	005532	1507*	
TST52	005672	1545*	
TST53	005776	1576*	
TST54	006044	1594*	
TST55	006140	1620*	
TST56	006206	1642*	
TST57	006332	1681*	
TST6	001660	570*	
TST60	006402	1702*	
TST61	006474	1713*	
TST62	006546	1720*	
TST63	006656	1733*	
TST64	007000	1841*	
TST65	007042	1860*	
TST66	007100	1877*	
TST67	007136	1893*	
TST7	001720	587*	
TST70	007232	1926*	
TST71	007340	1962*	
TST72	007412	1983*	
TST73	007464	2003*	
TST74	007536	2024*	
TST75	007610	2015*	
TST76	007662	2066*	
TST77	007756	2097*	
TTCR = 177564		202*	1504*
		3520*	3649*
		3662*	3672*
		3607*	3695*
		3711*	3757*
		3759	3771*
		3798*	
		3824*	3843*
		3855*	





AFPOP	259#	436	454	474	493	511	533	553	575	593	612	631	648	669	687
	706	730	748	767	787	804	825	844	865	883	902	922	939	960	978
	996	1020	1030	1057	1077	1094	1115	1134	1157	1175	1194	1214	1231	1252	1271
	1290	1305	1324	1344	1361	1382	1401	1421	1440	1459	1479	1495	1515	1532	1551
	1564	1582	1599	1612	1630	1652	1661	1689	1695	1711	1720	1741	1759	1769	1779
	1803	1812	1822	1847	1865	1881	1899	1913	1933	1943	1967	1971	1980	1992	2000
	2012	2029	2033	2050	2054	2071	2075	2085	2102	2106	2131	2140	2160	2169	2189
	2198	2218	2227	2247	2256	2296	2344	2354	2364	2374	2383	2394	2404	2414	2424
	2434	2444	2454	2464	2474	2484	2494	2504	2514	2524	2534	2544	2574	2583	2593
	2642	2624	2633	2643	2652	2667	2686	2703	2713	2722	2731	2740	2759	2768	2777
	2786	2795	2818	2824	2833	2842	2854	2863	2881	2890	2916	2934	2957	2964	2973
	2988	2997	3006	3030	3037	3046	3060	3069	3078	3101	3100	3117	3131	3140	3163
	3170	3179	3191	3200	3226	3234	3244	3253	3262	3271	3287	3296	3325	3333	3343
	3352	3361	3370	3386	3395	3419	3426	3435	3444	3453	3462	3470	3487	3506	3513
	3530	3537	3543	3561	3568	3574	3591	3600	3619	3626	3632	3655	3680	3700	3714
	3723	3732	3741	3761	3769	3804	3830	3840	3871	3893	3907	3920	3934	3940	3962
	3975	3989	4002	4010	4033	4052	4068	4084	4100	4116	4131	4142	4153	4164	4175
	4186	4197	4208	4227	4241	4255	4268	4280	4298	4296	4304	4317	4325	4333	4341
	4358	4374	4382	4390	4398	4413	4421	4429	4437	4469	4475	4484	4515	4524	4534
	4545	4553	4563												
APTFR0	257#	436	454	474	493	511	533	553	575	593	612	631	648	669	687
	706	730	748	767	787	804	825	844	865	883	902	922	939	960	978
	996	1020	1030	1057	1077	1094	1115	1134	1157	1175	1194	1214	1231	1252	1271
	1290	1305	1324	1344	1361	1382	1401	1421	1440	1459	1479	1495	1515	1532	1551
	1564	1582	1599	1612	1630	1652	1661	1689	1695	1711	1720	1741	1759	1769	1779
	1803	1812	1822	1847	1865	1881	1899	1913	1933	1943	1967	1971	1980	1992	2000
	2012	2029	2033	2050	2054	2071	2075	2085	2102	2106	2131	2140	2160	2169	2189
	2198	2218	2227	2247	2256	2296	2344	2354	2364	2374	2383	2394	2404	2414	2424
	2434	2444	2454	2464	2474	2484	2494	2504	2514	2524	2534	2544	2574	2583	2593
	2602	2624	2633	2643	2652	2667	2686	2703	2713	2722	2731	2740	2759	2768	2777
	2786	2795	2818	2824	2833	2842	2854	2863	2881	2890	2916	2934	2957	2964	2973
	2988	2997	3006	3030	3037	3046	3060	3069	3078	3101	3100	3117	3131	3140	3163
	3170	3179	3191	3200	3226	3234	3244	3253	3262	3271	3287	3296	3325	3333	3343
	3352	3361	3370	3386	3395	3419	3426	3435	3444	3453	3462	3470	3487	3506	3513
	3530	3537	3543	3561	3568	3574	3591	3600	3619	3626	3632	3655	3680	3700	3714
	3723	3732	3741	3761	3769	3804	3830	3840	3871	3893	3907	3920	3934	3940	3962
	3975	3989	4002	4010	4033	4052	4068	4084	4100	4116	4131	4142	4153	4164	4175
	4186	4197	4208	4227	4241	4255	4268	4280	4298	4296	4304	4317	4325	4333	4341
	4358	4374	4382	4390	4398	4413	4421	4429	4437	4469	4475	4484	4515	4524	4534
	4545	4553	4563												
H0R	261#	420	445	465	483	522	567	584	603	621	658	698	722	739	758
	777	814	857	874	893	912	949	980	1012	1029	1048	1067	1104	1149	1166
	1185	1204	1241	1282	1315	1334	1371	1413	1431	1450	1469	1504	1542	1573	1591
	1621	1639	1678	1699	1730	1749	1790	1830	1857	1874	1890	1923	1959	1980	2000
	2021	2042	2063	2094	2123	2152	2191	2210	2239	2270	2300	2557	2678	2749	2804
	2872	2890	2907	2925	2943	3016	3080	3150	3210	3309	3404	3496	3523	3553	3583
	3610	3643	3666	3690	3750	3781	3792	3815	3857	3883	4041	4125	4218	4447	4499
SFTTBI	259#	1707	1736	1755	3700	3709									
STARS	250#	307	317	344	346	353									
TMDR	260#	428	445	465	483	522	567	584	603	621	658	698	722	739	758
	777	814	857	874	893	912	949	980	1012	1029	1048	1067	1104	1149	1166
	1185	1204	1241	1282	1315	1334	1371	1413	1431	1450	1469	1504	1542	1573	1591
	1621	1639	1678	1699	1730	1749	1790	1830	1857	1874	1890	1923	1959	1980	2000
	2021	2042	2063	2094	2123	2152	2191	2210	2239	2270	2300	2557	2678	2749	2804
	2872	2890	2907	2925	2943	3016	3080	3150	3210	3309	3404	3496	3523	3553	3583
	3610	3643	3666	3690	3750	3781	3792	3815	3857	3883	4041	4125	4218	4447	4499

VTRP0	2115#	2123	2152	2191	2210	2239
.FACT1	300#	305				
.SAPT0	300#	315				
.SAPTH	300#	342				

.ABS. #21776 000

% ERRORS DETECTED: 0 HARD 17 SOFT  
 DEFAULT GLOBALS GENERATED: 0

DSKZ:DOKDBA,DSKZ:DOKOBA,SEQ/CRF/SOL=DSKZ:DOKDBA.P11  
 RUN-TIME: 14 52 1 SECONDS  
 RUN-TIME RATIO: 423/28=14.9  
 CORE USED: 10K (20 PAGES)