

KD11-K

PDP11/6X TRAPS TESTS
MD-11-DQKDB-A

EP-DQKDB-A-DL-A
COPYRIGHT © 1977
FICHE 1 OF 1

APR 1977
digital
MADE IN USA

Row	Col 1	Col 2	Col 3	Col 4	Col 5	Col 6	Col 7	Col 8	Col 9	Col 10
1
2
3
4
5
6
7
8
9
10

.REM %

IDENTIFICATION

PRODUCT CODE:	MAINDEC-11-DQKDB-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

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29

59
58
57
56
55
54
53
52
51
50
49
48
47
46
45
44
43
42
41
40
39
38
37
36
35
34
33
32
31
30
29
28
27
26
25
24
23
22
21
20
19
18
17
16
15
14
13
12
11
10
9
8
7
6
5
4
3
2
1

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 JRO" 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 JRO" 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 THE 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, DOT, 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

238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293

OPERATIONS.

9. ACT/APT COMPATABILITY

THE PROGRAM IS COMPATIBLE WITH ACT AND APT. UNDER APT, BEFORE HALTING ON ERROR, THE PROGRAM PASSES THE TEST NUMBER AND ERROR NUMBER TO LOCATIONS "\$TESTN" AND "\$FATAL" RESPECTIVELY.

```
%
;PDP-11/60 TRAP INSTRUCTION TEST AND ODD ON UNIQUE R6 OPERATIONS
;REVISED: SEPTEMBER 1976, JIM KAPADIA
.LIST ME ;LIST MACRO EXPANSIONS

.NLIST MC,MD,CND ;OR MACRO CALLS AND DEFINITIONS
.ENABL ABS,AMA ;ABSOLUTE BINARY OUTPUT
```

000000

TAB=%0

000001

LAST=%1

000001

SENO=1

000001

SSN=1

000002

FIRST=%2

010700

SCOPE=010700

;MOV REGISTER 7 TO REGISTER ZERO TO TAG LAST TEST

000000

HLT=HALT

000003

TRT=3

000003

BPT=3

000004

ITRAPS=4

;RESERVED INST AND ILLEGAL ADDRESSES

000004

ERRVEC=4

000014

TBITVEC=14

000030

EMTVEC=30

000014

TRTVEC=14

000014

BPTVEC=14

000020

IOTVEC=20

000034

TRAPVEC=34

177564

TTCSR=177564

177560

TRCSR=177560

177564

TPS=177564

177566

TPB=177566

000240

BELL=240

000240

LOOP=NOP

000240

NOP=240

177776

STATUS=177776

;A RESERVED INSTRUCTION
 ;RESERVED INSTRUCTION TRAP VECTOR

000007

RESINST=000007

000010

RESVEC=10

004700

ILLA=004700

000100

ILLB=100

H01

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 7
DOKDBA.P11 17-FEB-77 08:26

```

294          177776          CC=177776
295          000006          R6=%6
296          000005          RS=%5
297
298          076600          MED=076600          ;MAINT. EXAM & DEPOSIT
299
300
301
302          000200          .=200
303 000200 000137 001102  JMP      BEGIN
304          000300          .=300
305          .SBTTL  ACT11 HOOKS
306
307          ;*****
308          ;HOOKS REQUIRED BY ACT11
309          000300          $SVPC=.          ;SAVE PC
310          000046          .=46
311 000046 021520          $ENDAD          ;;1)SET LOC.46 TO ADDRESS OF $ENDAD IN .SEOP
312          000052          .=52
313 000052 000000          .WORD 0          ;;2)SET LOC.52 TO ZERO
314          000300          .=$SVPC          ;; RESTORE PC
315          .SBTTL  APT MAILBOX-ETABLE
316
317          ;*****
318          .EVEN
319          $MAIL:          ; APT MAILBOX
320          $MSGTY: .WORD  AMSGTY  ; MESSAGE TYPE CODE
321          $FATAL: .WORD  AFATAL  ; FATAL ERROR NUMBER
322          $TESTN: .WORD  ATESTN  ; TEST NUMBER
323          $PASS:  .WORD  APASS   ; PASS COUNT
324          $DEVCT: .WORD  ADEVCT  ; DEVICE COUNT
325          $UNIT:  .WORD  AUNIT   ; I/O UNIT NUMBER
326          $MSGAD: .WORD  AMSGAD  ; MESSAGE ADDRESS
327          $MSGLG: .WORD  AMSGLG  ; MESSAGE LENGTH
328          $ETABLE:          ; APT ENVIRONMENT TABLE
329          $ENV:   .BYTE  AENV    ; ENVIRONMENT BYTE
330          $ENVM: .BYTE  AENVM   ; ENVIRONMENT MODE BITS
331          $SWREG: .WORD  ASWREG  ; APT SWITCH REGISTER
332          $USWR:  .WORD  AUSWR   ; USER SWITCHES
333          $CPUOP: .WORD  ACPUOP  ; CPU TYPE, OPTIONS
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,PDQ=07,Q=10
337          ;*          BIT 10=REAL TIME CLOCK
338          ;*          BIT 9=FLOATING POINT PROCESSOR
339          ;*          BIT 8=MEMORY MANAGEMENT
340 000330          $ETEND:
341          .MEXIT
342          .SBTTL  APT PARAMETER BLOCK
343
344          ;*****
345          ;SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT
346          ;*****
347          000330          .SX=.          ;SAVE CURRENT LOCATION
348          000024          .=24          ;SET POWER FAIL TO POINT TO START OF PROGRAM
349 000024 000200          200          ;FOR APT START UP

```


350
351 000044 000044
352 000330 000330
353
354
355
356
357 000330
358 000330 000000
359 000332 000300
360 000334 000002
361 000336 000702
362 000340 000000
363 000342 000014
364
365 000304
366 000302
367
368

=44 ;: POINT TO APT INDIRECT ADDRESS PNTR.
\$APTHDR ;: POINT TO APT HEADER BLOCK
.=. \$X ;: RESET LOCATION COUNTER
:*****
; SETUP APT PARAMETER BLOCK AS DEFINED IN THE APT-PDP11 DIAGNOSTIC
; INTERFACE SPEC.
\$APTHD:
\$HIBTS: .WORD 0 ;: TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
\$MADDR: .WORD \$MAIL ;: ADDRESS OF APT MAILBOX (BITS 0-15)
\$STMT: .WORD 2 ;: RUN TIM OF LONGEST TEST
\$PASTM: .WORD 2 ;: RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
\$UNITM: .WORD 0 ;: ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
 .WORD \$ETEND-\$MAIL/2 ;: LENGTH MAILBOX-ETABLE (WORDS)
\$STNM=\$TESTN
\$ERRR=\$FATAL

J01

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 9
DQKDBA.P11 17-FEB-77 08:26 APT PARAMETER BLOCK

```

369          001000          .=1000
370
371
372 001000 000000          K1: 0
373 001002 000000          K2: 0
374 001004 000000          K3: 0
375 001006 000000          K4: 0
376 001010 000000          K5: 0
377 001012 000000          K6: 0
378 001014 052525          K7: 52525
379 001016 052400          K10: 52400
380 001020 000000          K11: 0
381 001022 000000          K12: 0
382
383 001024 000000          NAMFLG: 0
384
385
386 001026 005015 040515 047111 MSGNAM: .ASCIZ <15><12>.MAINDEC-11-DQKDB-A PDP 11/6X TRAPS TEST.
387 001034 042504 026503 030461
388 001042 042055 045521 041104
389 001050 040755 020040 042120
390 001056 020120 030461 033057
391 001064 020130 051124 050101
392 001072 020123 042524 052123
393 001100          000
394
395          001102          .EVEN
396
397 001102 012737 177777 021546 BEGIN: MOV    #-1,2#PASSPT ;CLEAR THE ITERATION COUNTER
398
399
400
401 001110 005037 000300          BEGIN1: CLR    $MSGTY
402 001114 012737 021570 000024          MOV    #PWRDWN,24 ;SET UP THE POWER DOWN VECTOR
403 001122 012737 000340 000026          MOV    #340,26 ;SET UP POWER DOWN PRIORITY
404 001130 005037 000304          CLR    $STNM
405 001134 005037 000302          CLR    $ERROR
406 001140 012702 000300          MOV    #MSGTY,R2
407 001144 012703 000302          MOV    #FATAL,R3
408 001150 012706 021774          MOV    #BUFF,%6 ;SET UP STACK POINTER
409
410          ;PRINT MAINDEC NUMBER AND NAME ONLY THE FIRST TIME
411
412 001154 005737 001024          TST    NAMFLG ;NAME PRINTED NCE?
413 001160 001021          BNE    BGN3 ;IF YES DONT PRINT AGAIN
414 001162 005237 001024          INC    NAMFLG ;SET FLAG
415 001166 132737 000040 000321          BITB  #40,$ENVM ;WILL APT ALLOW PRINTING?
416 001174 001013          BNE    BGN3 ;NO
417 001176 012700 001026          MOV    #MSGNAM,R0
418 001202 105737 177564          BGN2: TSTB  2#TPS ;TTY READY?
419 001206 100375          BPL    BGN2
420 001210 112037 177566          MOVB  (R0)+,2#TPB ;PRINT CHARACTER
421 001214 001372          BNE    BGN2 ;PRINT NEXT ONE IF NOT DONE
422
423 001216 105737 177564          BGN2A: TSTB  2#TPS ;WAIT FOR DONE
424 001222 100375          BPL    BGN2A

```

K01

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 10
 DOKDBA.P11 17-FEB-77 08:26 APT PARAMETER BLOCK

```

425 001224          BGN3:
426
427 001224 010700          SCOPE
428 ;*****
429 ;*TEST 1 TEST THAT A TRAP OCCURS ON A RESERVED INSTRUCTION (76000)
430 ;*****
431 001226 005237 000304    TST1:  INC  2*STESTN
432 001232 012706 021774    MOV  #BUFF,%6                ;STACK POINTER SETUP
433 001236 012737 001264    MOV  #RETA,RESVEC           ;LOAD RESERVED INST. TRAP VECTOR
434 001244 005037 000012    CLR  RESVEC+2              ;AND STATUS
435 001250 000007          RESINST ;RESERVED INSTRUCTION (SHOULT TRAP)
436
437 001252 005212          INC  (R2)                   ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
438 001254 012713 000001    MOV  #1,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
439 001260 000000          HLT  ;ERROR! RESERVED INSTRUCTION FAILED TO TRAP
440 001262 000240          LOOP ;REPLACE THIS INSTRUCTION BY A
441 ;"JMP 2R0" (000110) TO GET A
442 ;SCOPE LOOP AND HIT CONTINUE
443 001264 010700          RETA:  SCOPE
444
445 ;*****
446 ;*TEST 2 TEST DECREMENT OF STACK POINTER ON A RESERVED INSTRUCTION TRAP
447 ;*****
448 001266 005237 000304    TST2:  INC  2*STESTN
449 001272 012706 021774    MOV  #BUFF,%6                ;STACK POINTER SETUP
450 001276 012737 001306    MOV  #RETB,RESVEC           ;RETURN POINTER
451 001304 000007          RESINST ;DO A RESERVED INSTRUCTION
452 001306 020627 021770    RETB:  CMP  %6,#BUFF-4         ;TEST DECREMENT OF %6
453 001312 001405          BEQ  1$
454
455 001314 005212          INC  (R2)                   ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
456 001316 012713 000002    MOV  #2,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
457 001322 000000          HLT  ;STACK POINTER (R6) WAS NOT PUSHED
458 ;DOWN BY TWO WORDS WHEN A RESERVED
459 ;INSTRUCTION TRAPPED
460 001324 000240          LOOP ;REPLACE THIS INSTRUCTION BY A
461 ;"JMP 2R0" (000110) TO GET A
462 ;SCOPE LOOP AND HIT CONTINUE
463 001326 010700          1$:  SCOPE
464
465 ;*****
466 ;*TEST 3 TEST THAT PROPER P.C. IS SAVED ON A RESERVED INSTRUCTION TRAP
467 ;*****
468 001330 005237 000304    TST3:  INC  2*STESTN
469 001334 012706 021774    MOV  #BUFF,%6                ;STACK POINTER SETUP
470 001340 012737 001350    MOV  #RETC,RESVEC
471 001346 000007          INSTC: RESINST ;TRAP ON THIS INSTRUCTION
472 001350 022737 001350    RETC:  CMP  #,BUFF-4         ;CHECK FOR INCREMENTED P.C.
473 001356 001405          BEQ  1$
474
475 001360 005212          INC  (R2)                   ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
476 001362 012713 000003    MOV  #3,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
477 001366 000000          HLT  ;WRONG "OLD" PC WAS SAVED ON THE STACK
478 ;WHEN A RESERVED INSTR. TRAPPED
479 001370 000240          LOOP ;REPLACE THIS INSTRUCTION BY A
480 ;"JMP 2R0" (000110) TO GET A
  
```

L01

T3 TEST THAT PROPER P.C. IS SAVED ON A RESERVED INSTRUCTION TRAP

```

481                                     ;SCOPE LOOP AND HIT CONTINUE
482 001372 010700 1S: SCOPE
483                                     ;*****
484                                     ;*TEST 4 TEST THAT "OLD" CC AND PRIORITY ARE SAVED ON A RESERVED INSTR. TRAP
485                                     ;*****
486 001374 005237 000304 TST4: INC 2*STESTN
487 001400 012706 021774 MOV #BUFF,%6 ;SET UP
488 001404 012737 001420 000010 MOV #RETD,RESVEC
489 001412 005037 177776 CLR CC ;CLEAR CC AND PRIORITY
490 001416 000007 RESINST ;TRAP ON RESERVED INSTRUCTION
491 001420 013700 021772 RETD: MOV BUFF-2,%0 ;GET SAVED STATUS & TEST FOR ALL 0'S
492 001424 001405 BEQ 1S ;BRANCH IF 0
493
494 001426 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
495 001430 012713 000004 MOV #4,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
496 001434 000000 HLT ;WRONG "OLD" PSW SAVED ON STACK
497 ;WHEN A RESERVED INSTR. TRAPPED
498 ;EXPECT OLD PSW=0
499 001436 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
500 ;"JMP 2RD" (000110) TO GET A
501 ;SCOPE LOOP AND HIT CONTINUE
502 001440 010700 1S: SCOPE
503
504 001442 012706 021774 MOV #BUFF,%6 ;INITIALIZE THE STACK POINTER
505 001446 012737 001464 000010 MOV #RETE,RESVEC
506 001454 012737 000357 177776 MOV #357,CC
507 001462 000007 RESINST ;PRE SET THE STATUS WORD
508 001464 013700 021772 RETE: MOV BUFF-2,%0 ;RESERVED INSTRUCTION TRAP
509 001470 022700 000357 CMP #357,%0 ;GET SAVED STATUS
510 001474 001405 BEQ 1S ;SAVED STATUS CORRECT?
511 ;BRANCH IF CORRECT
512 001476 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
513 001500 012713 000005 MOV #5,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
514 001504 000000 HLT ;WRONG "OLD" PSW SAVED ON STACK
515 ;WHEN A RESERVED INSTR. TRAPPED
516 ;EXPECT OLD PSW=357
517 001506 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
518 ;"JMP 2RD" (000110) TO GET A
519 ;SCOPE LOOP AND HIT CONTINUE
520 001510 010700 1S: SCOPE
521
522 ;*****
523 ;*TEST 5 TEST THAT "NEW" STATUS IS CORRECT ON A RESERVED INSTRUCTION TRAP
524 ;*****
525 001512 005237 000304 TST5: INC 2*STESTN
526 001516 012706 021774 MOV #BUFF,%6 ;INITIALIZE THE STACK POINTER
527 001522 012737 001544 000010 MOV #RETF,RESVEC ;SET UP VECTOR
528 001530 005037 000012 177776 CLR RESVEC+2 ;CLEAR 'NEW' STATUS
529 001534 012737 000357 177776 MOV #357,CC ;PRE SET THE STATUS WORD
530 001542 000007 RESINST ;DO A RESERVED INSTRUCTION
531 001544 013700 177776 RETF: MOV CC,%0 ;GET & TEST THE 'NEW' STATUS WORD
532 001550 001405 BEQ 1S ;BRANCH IF ALL 0'S
533
534 001552 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
535 001554 012713 000006 MOV #6,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
536 001560 000000 HLT ;"NEW" PSW WAS INCORRECT WHEN A

```

MO1

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 12
 DOKDBA.P11 17-FEB-77 08:26 TS TEST THAT "NEW" STATUS IS CORRECT ON A RESERVED INSTRUCTION TRAP

```

537                                     ;RESERVED INSTR. TRAPPED
538                                     ;EXPECT NEW PSW=0
539 001562 000240                       LOOP                               ;REPLACE THIS INSTRUCTION BY A
540                                     ;"JMP 3RD" (000110) TO GET A
541                                     ;SCOPE LOOP AND HIT CONTINUE
542 001564 005037 177776                 1$: CLR CC
543 001570 010700                       SCOPE
544
545 001572 012706 021774                 MOV #BUFF,%6 ;SET UP THE STACK POINTER
546 001576 012737 001620 000010         MOV #RETG,RESVEC
547 001604 012737 000357 000012         MOV #357,RESVEC+2 ;LOAD THE NEW STATUS WORD
548 001612 005037 177776                 CLR CC ;PRE SET THE STATUS WORD
549 001616 000037                       RESINST ;DO A RESERVED INSTRUCTION
550 001620 013700 177776                 RETG: MOV CC,%0 ;GET THE 'NEW' STATUS WORD
551 001624 022700 000357                 CMP #357,%0 ;WAS 'NEW' STATUS CORRECTLY LOADED
552 001630 001405                       BEQ 1$ ;BRANCH IF CORRECT
553
554 001632 005212                       INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
555 001634 012713 000007                 MOV #7,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
556 001640 000000                       HLT ;"NEW" PSW WAS INCORRECT WHEN A
557                                     ;RESERVED INSTRUCTION TRAPPED
558                                     ;EXPECT NEW PSW=357
559 001642 000240                       LOOP                               ;REPLACE THIS INSTRUCTION BY A
560                                     ;"JMP 3RD" (000110) TO GET A
561                                     ;SCOPE LOOP AND HIT CONTINUE
562
563 001644 012737 000012 000010         1$: MOV #RESVEC+2,RESVEC ;RESTORE RESERVED INSTRUCTION
564 001652 005037 000012                 CLR RESVEC+2 ;TO HALT AT RESVEC+2
565 001656 010700                       SCOPE
566
567                                     ;*****
568                                     ;*TEST 6 TEST THAT A TRAP OCCURES FOR A "TRAP" INSTRUCTION
569                                     ;*****
570 001660 005237 000304                 TST6: INC #STESTN
571 001664 012706 021774                 MOV #BUFF,%6 ;STACK POINTER SETUP
572 001670 012737 001716 000034         MOV #RETA1,TRAPVEC ;LOAD TRAP VECTOR
573 001676 005037 000036                 CLR TRAPVEC+2
574 001702 104400                       TRAP ;DO A TRAP INSTRUCTION
575
576 001704 005212                       INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
577 001706 012713 000010                 MOV #10,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
578 001712 003000                       HLT ;"TRAP" INSTRUCTION DID NOT TRAP
579 001714 000240                       LOOP ;REPLACE THIS INSTRUCTION BY A
580                                     ;"JMP 3RD" (000110) TO GET A
581                                     ;SCOPE LOOP AND HIT CONTINUE
582 001716 010700                       RETA1: SCOPE
583
584                                     ;*****
585                                     ;*TEST 7 TEST DECREMENT OF STACK POINTER ON A "TRAP" INSTRUCTION
586                                     ;*****
587 001720 005237 000304                 TST7: INC #STESTN
588 001724 012706 021774                 MOV #BUFF,%6 ;STACK POINTER SETUP
589 001730 012737 001740 000034         MOV #RETB1,TRAPVEC ;RETURN POINTER
590 001736 104400                       TRAP ;DO A TRAP INSTRUCTION
591 001740 020627 021770                 RETB1: CMP %6,#BUFF-4 ;TEST DECREMENT OF %6
592 001744 001405                       BEQ 1$
  
```

NO1

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 13
 DOKDBA.P11 17-FEB-77 08:26 T7 TEST DECREMENT OF STACK POINTER ON A "TRAP" INSTRUCTION

```

593
594 001746 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
595 001750 012713 000011  MOV      #11,(R3)      ;MOVE TO MAIL BOX ($FATAL) THE ERROR
596 001754 000000          HLT                    ;STACK POINTER (R6) DID NOT DECREMENT
597                                     ;TWO WORDS WHEN "TRAP" TRAPPED
598 001756 000240          LOOP                   ;REPLACE THIS INSTRUCTION BY A
599                                     ;"JMP 2R0" (000110) TO GET A
600                                     ;SCOPE LOOP AND HIT CONTINUE
601 001760 010700          1$:      SCOPE
602
603                                     ;*****
604                                     ;*TEST 10 TEST THAT PROPER P.C. IS SAVED ON A "TRAP" INSTRUCTION
605                                     ;*****
606 001762 005237 000304    TST10:  INC      @#STESTN
607 001766 012706 021774    MOV      #BUFF,%6      ;STACK POINTER SETUP
608 001772 012737 002002 000034  MOV      #RETC1,TRAPVEC ;RETURN FROM TRAP POINTER
609 002000 104400          TRAP                   ;TRAP ON THIS INSTRUCTION
610 002002 022737 002002 021770  RETC1:  CMP      #.,BUFF-4    ;CHECK INCREMENTED P.C.
611 002010 001405          BEQ      1$
612
613 002012 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
614 002014 012713 000012  MOV      #12,(R3)      ;MOVE TO MAIL BOX ($FATAL) THE ERROR
615 002020 000000          HLT                    ;WRONG "OLD" PC SAVED ON THE STACK
616                                     ;WHEN "TRAP" TRAPPED
617 002022 000240          LOOP                   ;REPLACE THIS INSTRUCTION BY A
618                                     ;"JMP 2R0" (000110) TO GET A
619                                     ;SCOPE LOOP AND HIT CONTINUE
620 002024 010700          1$:      SCOPE
621
622                                     ;*****
623                                     ;*TEST 11 TEST THAT "OLD" CC & PRIORITY ARE PLACED ON STACK, ON A "TRAP" TRAP
624                                     ;*****
625 002026 005237 000304    TST11:  INC      @#STESTN
626 002032 012706 021774    MOV      #BUFF,%6      ;SET UP
627 002036 012737 002052 000034  MOV      #RETD1,TRAPVEC ;SET UP
628 002044 005037 177776    CLR      CC             ;CLEAR CC AND PRIORITY
629 002050 104400          TRAP                   ;DO A TRAP INSTRUCTION
630 002052 013700 021772    RETD1:  MOV      BUFF-2,%0 ;GET & TEST THE SAVED STATUS
631 002056 001405          BEQ      1$            ;BRANCH IF ALL 0'S
632
633 002060 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
634 002062 012713 000013  MOV      #13,(R3)      ;MOVE TO MAIL BOX ($FATAL) THE ERROR
635 002066 000000          HLT                    ;WRONG "OLD" PSW SAVED ON THE STACK
636                                     ;WHEN "TRAP" TRAPPED, EXPECT "0"
637 002070 000240          LOOP                   ;REPLACE THIS INSTRUCTION BY A
638                                     ;"JMP 2R0" (000110) TO GET A
639                                     ;SCOPE LOOP AND HIT CONTINUE
640 002072 010700          1$:      SCOPE
641
642 002074 012706 021774    MOV      #BUFF,%6      ;INITIALIZE THE STACK POINTER
643 002100 012737 002116 000034  MOV      #RETE1,TRAPVEC ;SET UP
644 002106 012737 000357 177776  MOV      #357,CC       ;PRE SET THE STATUS WORD
645 002114 104400          TRAP                   ;DO A TRAP INSTRUCTION
646 002116 013700 021772    RETE1:  MOV      BUFF-2,%0 ;GET THE SAVED STATUS WORD
647 002122 022700 000357    CMP      #357,%0      ;WAS CORRECT STATUS SAVED ON THE STACK
648 002126 001405          BEQ      1$            ;BRANCH IF CORRECT

```


.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 15
 DOKDBA.P11 17-FEB-77 08:26

T13 TEST THAT ALL COMBINATION OF "TRAP" WILL CAUSE A TRAP

705	002330	104400			RB1:	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	000017			MOV	#17,(R3)	;MOVE TO MAIL BOX (\$FATAL) THE ERROR
709	002340	000000				HLT		; "TRAP" INSTRUCTION HELD IN "RB1"
710								;FAILED TO TRAP
711	002342	000240				LOOP		;REPLACE THIS INSTRUCTION BY A
712								; "JMP 2RO" (00C110) TO GET A
713								;SCOPE LOOP AND HIT CONTINUE
714	002344	005237	002330		RA1:	INC	RB1	;INCREMENT TRAP INSTRUCTION
715	002350	022737	104777	002330		CMP	#104777,RB1	;TRAP+377 TO UPPER LIMIT
716	002356	103362				BHIS	RC1	;HAVE WE TESTED ALL
717								;YES
718								
719	002360	012737	000036	000034		MOV	#36,TRAPVEC	;RESTORE TRAP VECTOR TO
720	002366	005037	000036			CLR	TRAPVEC+2	;HALT AT 36
721	002372	010700				SCOPE		


```

722 ;*****
723 ;*TEST 14 TEST THAT A TRAP OCCURES ON AN "IOT" INSTRUCTION
724 ;*****
725 002374 005237 000304 TST14: INC @#TESTN
726 002400 012706 021774 MOV #BUFF,%6 ;STACK POINTER SETUP
727 002404 012737 002432 000020 MOV #RETA2,IOTVEC ;RETURN LOCATION
728 002412 005037 000022 CLR IOTVEC+2
729 002416 000004 IOT
730
731 002420 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
732 002422 012713 000020 MOV #20,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
733 002426 000000 HLT ;"IOT" INSTRUCTION FAILED TO TRAP
734 002430 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
735 ;"JMP @RO" (000110) TO GET A
736 ;SCOPE LOOP AND HIT CONTINUE
737 002432 010700 RETA2: SCOPE
738
739 ;*****
740 ;*TEST 15 TEST DECREMENT OF STACK POINTER ON AN "IOT" TRAP
741 ;*****
742 002434 005237 000304 TST15: INC @#TESTN
743 002440 012706 021774 MOV #BUFF,%6 ;STACK POINTER SETUP
744 002444 012737 002454 000020 MOV #RETB2,IOTVEC ;RETURN POINTER
745 002452 000004 IOT
746 002454 020627 021770 RETB2: CMP %6,#BUFF-4 ;TEST DECREMENT OF %6
747 002460 001405 BEQ IS
748
749 002462 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
750 002464 012713 000021 MOV #21,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
751 002470 000000 HLT ;STACK POINTER DID NOT DECREMENT BY
752 ;TWO WORDS ON AN "IOT" TRAP
753 002472 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
754 ;"JMP @RO" (000110) TO GET A
755 ;SCOPE LOOP AND HIT CONTINUE
756 002474 010700 IS: SCOPE
757
758 ;*****
759 ;*TEST 16 TEST THAT PROPER P.C. IS SAVED ON AN "IOT" TRAP
760 ;*****
761 002476 005237 000304 TST16: INC @#TESTN
762 002502 012706 021774 MOV #BUFF,%6 ;STACK POINTER SETUP
763 002506 012737 002516 000020 MOV #RETC2,IOTVEC ;RETURN FROM TRAP POINTER
764 002514 000004 IOT ;TRAP ON THIS INSTRUCTION
765 002516 022737 002516 021770 RETC2: CMP #.,BUFF-4 ;CHECK FOR INCREMENTED P.C.
766 002524 001405 BEQ IS
767
768 002526 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
769 002530 012713 000022 MOV #22,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
770 002534 000000 HLT ;WRONG "OLD" PC WAS SAVED ON
771 ;STACK WHEN "IOT" TRAPPED
772 002536 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
773 ;"JMP @RO" (000110) TO GET A
774 ;SCOPE LOOP AND HIT CONTINUE
775 002540 010700 IS: SCOPE
776
777 ;*****
  
```

E02

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 17
 DOKDBA.P11 17-FEB-77 08:26

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 002542 005237 000304 TST17: INC @STESTN
781 002546 012706 021774 MOV #BUFF,%6 ;SET UP
782 002552 012737 002566 000020 MOV #RETD2,IOTVEC ;SET UP
783 002560 005037 177776 CLR IOTVEC ;CLEAR CC AND PRIORITY
784 002564 000004 IOT ;TRAP
785 002566 013700 021772 RETD2: MOV BUFF-2,%0 ;GET & TEST SAVED STATUS
786 002572 001405 BEQ IS
787
788 002574 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" PSW SAVED ON STACK
791 ;WHEN "IOT" TRAPPED, EXPECT 0
792 002604 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
793 ;"JMP @RO" (000110) TO GET A
794 ;SCOPE LOOP AND HIT CONTINUE
795 002606 010700 IS: SCOPE
796
797 002610 012706 021774 MOV #BUFF,%6 ;SET UP
798 002614 012737 002632 000020 MOV #RETE2,IOTVEC ;SET UP
799 002622 012737 000357 177776 MOV #357,CC ;PRE SET STATUS
800 002630 000004 IOT
801 002632 013700 021772 RETE2: MOV BUFF-2,%0 ;GET SAVED STATUS
802 002636 022700 000357 CMP #357,%0 ;SAVED STATUS CORRECT?
803 002642 001404 BEQ IS ;BRANCH IF CORRECT
804
805 002644 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
806 002646 012713 000024 MOV #24,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
807 002652 000000 HLT ;WRONG "OLD" PSW SAVED ON STACK
808 ;WHEN "IOT" TRAPPED, EXPECT 357
809 002654 000240 IS: LOOP ;REPLACE THIS INSTRUCTION BY A
810 ;"JMP @RO" (000110) TO GET A
811 ;SCOPE LOOP AND HIT CONTINUE
812 002656 010700 SCOPE
813
814 ;*****
815 ;*TEST 20 TEST THAT "NEW" STATUS IS CORRECT ON AN "IOT" TRAP
816 ;*****
817 002660 005237 000304 TST20: INC @STESTN
818 002664 012706 021774 MOV #BUFF,%6
819 002670 012737 002712 000020 MOV #RETF2,IOTVEC
820 002676 005037 000022 CLR IOTVEC+2 ;CLEAR FUTURE PRIORITY AND CC
821 002702 012737 030357 177776 MOV #30357,CC ;PRE SET STATUS
822 002710 000004 IOT
823 002712 013700 177776 RETF2: MOV CC,%0 ;GET & TEST 'NEW' STATUS
824 002716 001405 BEQ IS ;BRANCH IF CORRECT
825
826 002720 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" PSW LOADED WHEN
829 ;"IOT" TRAPPED, EXPECT 0
830 002730 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
831 ;"JMP @RO" (000110) TO GET A
832 ;SCOPE LOOP AND HIT CONTINUE
833 002732 005037 177776 IS: CLR CC
  
```

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

834	002736	010700			SCOPE		
835							
836	002740	012706	021774		MOV	#BUFF,%6	
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,%0	;GET THE 'NEW' STATUS
842	002772	022700	000357		CMP	#357,%0	;IS 'NEW' STATUS CORRECT
843	002776	001405			BEQ	1\$;BRANCH IF CORRECT
844							
845	003000	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" PSW LOADED WHEN
848							; "IOT" TRAPPED, EXPECT 357
849	003010	000240			LOOP		;REPLACE THIS INSTRUCTION BY A
850							; "JMP 2RD" (000110) TO GET A
851							; SCOPE LOOP AND HIT CONTINUE
852	003012				1\$:		
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		

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 19
 DOKDBA.P11 17-FEB-77 08:26

T21 TEST THAT A TRAP OCCURS ON AN "EMT" INSTRUCTION

```

857 ;*****
858 ;*TEST 21 TEST THAT A TRAP OCCURS ON AN "EMT" INSTRUCTION
859 ;*****
860 003026 005237 000304 TST21: INC 2*STESTN
861 003032 012706 021774 MOV #BUFF,%6 ;STACK POINTER SETUP
862 003036 012737 003064 000030 MOV #RETA3,EMTVEC ;RETURN LOCATION
863 003044 005037 000032 CLR EMTVEC+2
864 003050 104000 EMT
865
866 003052 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
867 003054 012713 000027 MOV #27,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
868 003060 000000 HLT ;"EMT" FAILED TO TRAP
869 003062 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
870 ;"JMP 2R0" (000110) TO GET A
871 ;SCOPE LOOP AND HIT CONTINUE
872 003064 010700 RETA3: SCOPE
873
874 ;*****
875 ;*TEST 22 TEST DECREMENT OF STACK POINTER ON A "EMT" TRAP
876 ;*****
877 003066 005237 000304 TST22: INC 2*STESTN
878 003072 012706 021774 MOV #BUFF,%6 ;STACK POINTER SETUP
879 003076 012737 003106 000030 MOV #RETB3,EMTVEC ;RETURN POINTER
880 003104 104000 EMT
881 003106 020627 021770 RETB3: CMP %6,#BUFF-4 ;TEST DECREMENT OF %6
882 003112 001405 BEQ 1$
883
884 003114 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
885 003116 012713 000030 MOV #30,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
886 003122 000000 HLT ;STACK POINTER DID NOT DECREMENT BY
887 ;TWO WORDS ON "EMT" TRAP
888 003124 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
889 ;"JMP 2R0" (000110) TO GET A
890 ;SCOPE LOOP AND HIT CONTINUE
891 003126 010700 1$: SCOPE
892
893 ;*****
894 ;*TEST 23 TEST THAT PROPER P.C IS SAVED ON A "EMT" TRAP
895 ;*****
896 003130 005237 000304 TST23: INC 2*STESTN
897 003134 012706 021774 MOV #BUFF,%6 ;STACK POINTER SETUP
898 003140 012737 003150 000030 MOV #RETC3,EMTVEC ;RETURN FROM TRAP POINTER
899 003146 104000 EMT ;TRAP ON THIS INSTRUCTION
900 003150 022737 003150 021770 RETC3: CMP #,BUFF-4 ;CHECK FOR INCREMENTED P.C.
901 003156 001405 BEQ 1$
902
903 003160 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
904 003162 012713 000031 MOV #31,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
905 003166 000000 HLT ;WRONG "OLD" PC SAVED ON STACK
906 ;ON "EMT" TRAP
907 003170 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
908 ;"JMP 2R0" (000110) TO GET A
909 ;SCOPE LOOP AND HIT CONTINUE
910 003172 010700 1$: SCOPE
911
912 ;*****

```

H02

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 20
DOKDBA.P11 17-FEB-77 08:26

T24 TEST THAT "OLD" CC AND PRIORITY ARE PLACED ON STACK ON A "EMT" TRAP

```

913 ;*TEST 24 TEST THAT "OLD" CC AND PRIORITY ARE PLACED ON STACK ON A "EMT" TRAP
914 ;*****
915 003174 005237 000304 TST24: INC @#TESTN
916 003200 012706 021774 MOV #BUFF,%6 ;SET UP
917 003204 012737 003220 000030 MOV #RETD3,EMTVEC ;SET UP
918 005212 005037 177776 CLR CC ;CLEAR CC AND PRIORITY
919 003216 104000 EMT TRAP
920 003220 013700 021772 RETD3: MOV BUFF-2,%0 ;GET THE SAVED STATUS,
921 003224 001405 BEQ IS ;BRANCH IF SAVED STATUS IS CORRECT
922
923 003226 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
924 003230 012713 000032 MOV #32,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
925 003234 000000 HLT ;WRONG "OLD" PSW SAVED ON
926 ;STACK ON "EMT" TRAP, EXPECT 0
927 003236 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
928 ;"JMP @RO" (000110) TO GET A
929 ;SCOPE LOOP AND HIT CONTINUE
930 003240 010700 IS: SCOPE
931
932 003242 012706 021774 MOV #BUFF,%6 ;SET UP
933 003246 012737 003264 000030 MOV #RETE3,EMTVEC ;SET UP
934 003254 012737 000357 177776 MOV #357,CC ;PRE SET STATUS
935 003262 104000 EMT
936 003264 013700 021772 RETE3: MOV BUFF-2,%0 ;GET THE SAVED STATUS
937 003270 022700 000357 CMP #357,%0 ;SAVED STATUS CORRECT?
938 003274 001405 BEQ IS ;BRANCH IF CORRECT
939
940 003276 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
941 003300 012713 000033 MOV #33,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
942 003304 000000 HLT ;WRONG "OLD" PSW SAVED ON STACK
943 ;ON "EMT" TRAP, EXPECT 357
944 003306 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
945 ;"JMP @RO" (000110) TO GET A
946 ;SCOPE LOOP AND HIT CONTINUE
947 003310 010700 IS: SCOPE
948
949 ;*****
950 ;*TEST 25 TEST THAT "NEW" STATUS IS CORRECT ON A "EMT" TRAP
951 ;*****
952 003312 005237 000304 TST25: INC @#TESTN
953 003316 012706 021774 MOV #BUFF,%6
954 003322 012737 003344 000030 MOV #RETF3,EMTVEC
955 003330 005037 000032 CLR EMTVEC+2 ;CLEAR FUTURE PRIORITY AND CC
956 003334 012737 000357 177776 MOV #357,CC ;PRE SET THE STATUS WORD
957 003342 104000 EMT
958 003344 013700 177776 RETF3: MOV CC,%0 ;GET THE 'NEW' STATUS
959 003350 001405 BEQ IS ;BRANCH IF 'NEW' STATUS IS CORRECT
960
961 003352 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
962 003354 012713 000034 MOV #34,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
963 003360 000000 HLT ;WRONG "NEW" PSW LOADED ON
964 ;"EMT" TRAP, EXPECT 0
965 003362 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
966 ;"JMP @RO" (000110) TO GET A
967 ;SCOPE LOOP AND HIT CONTINUE
968 003364 010700 IS: SCOPE

```

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 21
 DOKDBA.P11 17-FEB-77 08:26 T25 TEST THAT "NEW" STATUS IS CORRECT ON A "EMT" TRAP

```

969
970 003366 012706 021774          MOV    #BUFF,%6
971 003372 012737 003414 000030    MOV    #RETG3,EMTVEC
972 003400 012737 000357 000032    MOV    #357,EMTVEC+2 ;LOAD 'NEW' STATUS
973 003406 005037 177776          CLR    CC              ;PRE SET THE STATUS
974 003412 104000          EMT
975 003414 013700 177776          RETG3: MOV   CC,%0        ;GET THE 'NEW' STATUS
976 003420 022700 000357          CMP    #357,%0        ;IS IT CORRECT
977 003424 001405          BEQ    1$             ;BRANCH IF CORRECT
978
979 003426 005212          INC    (R2)           ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
980 003430 012713 000035          MOV    #35,(R3)      ;MOVE TO MAIL BOX ($FATAL) THE ERROR
981 003434 000000          HLT
982
983 003436 000240          LOOP
984
985
986 003440 010700          1$:   SCOPE
987
988 ;*****
989 ;*TEST 26 TEST THAT ALL COMBINATION OF EMT WILL CAUSE A TRAP
990 ;*****
991 003442 005237 000304          TST26: INC    @#STESTN
992 003446 012737 104000 003466    MOV    #EMT,RB        ;INITIALIZE BASE EMT INSTRUCTION
993 003454 012737 003502 000030    MOV    #RA,30         ;RETURN FROM TRAP TO RA
994 003462 012706 021774          RC:   MOV    #BUFF,%6 ;SET UP STACK POINTER
995 003466 104000          RB:   EMT              ;TRAP INST, WILL BE MODIFIED TO EMT+377
996
997 003470 005212          INC    (R2)           ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
998 003472 012713 000036          MOV    #36,(R3)      ;MOVE TO MAIL BOX ($FATAL) THE ERROR
999 003476 000000          HLT
1000 ;"EMT" INSTR. CONTAINED IN "RB" ABOVE,
1001 ;FAILED TO TRAP
1002 ;REPLACE THIS INSTRUCTION BY A
1003 ;"JMP 2RO" (000110) TO GET A
1004 ;SCOPE LOOP AND HIT CONTINUE
1004 003502 005237 003466          RA:   INC    RB        ;INCREMENT TRAP INSTRUCTION
1005 003506 022737 104377 003466    CMP    #104377,RB    ;EMT+377 TO EMT?
1006 003514 103362          BHS   RC              ;HAVE WE TESTED ALL
1007 ;YES
1008 003516 012737 000032 000030    MOV    #32,EMTVEC    ;RESTORE EMT TRAP TO HALT
1009 003524 005037 000032          CLR    EMTVEC+2      ;AT 32
1010 003530 010700          SCOPE
  
```

```

1011
1012
1013 ;*****
1014 ;*TEST 27 TEST THAT A TRAP OCCURS ON A "BPT" (TRACE TRAP)
1015 ;*****
1015 003532 005237 000304 TST27: INC @#TESTN
1016 003536 012706 021774 MOV #BUFF,%6 ;STACK POINTER SETUP
1017 003542 012737 003570 000014 MOV #RETA4,BPTVEC ;RETURN LOCATION
1019 003550 005037 000016 CLR BPTVEC+2
1019 003554 000003 BPT ;TRACE TRAP
1021 003556 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1022 003560 012713 000037 MOV #37,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1023 003564 000000 HLT
1024 003566 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
1025 ;"JMP @RO" (000110) TO GET A
1026 ;SCOPE LOOP AND HIT CONTINUE
1027 003570 010700 RETA4: SCOPE ;"BPT" FAILED TO TRAP
1028
1029 ;*****
1030 ;*TEST 30 TEST DECREMENT OF STACK POINTER ON A "BPT" TRAP
1031 ;*****
1032 003572 005237 000304 TST30: INC @#TESTN
1033 003576 012706 021774 MOV #BUFF,%6 ;STACK POINTER SETUP
1034 003602 012737 003612 000014 MOV #RETB4,BPTVEC ;RETURN POINTER
1035 003610 000003 BPT ;RESERVED INSTRUCTION
1036 003612 020627 021770 RETB4: CMP %6,#BUFF-4 ;TEST DECREMENT OF %6
1037 003616 001405 BEQ 1$
1038
1039 003620 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1040 003622 012713 000040 MOV #40,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1041 003626 000000 HLT ;STACK POINTER DID NOT DECREMENT BY 2
1042 ;WORDS ON "BPT" TRAP
1043 003630 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
1044 ;"JMP @RO" (000110) TO GET A
1045 ;SCOPE LOOP AND HIT CONTINUE
1046 003632 010700 1$: SCOPE
1047
1048 ;*****
1049 ;*TEST 31 TEST THAT PROPER P.C. IS SAVED ON A "BPT" TRAP
1050 ;*****
1051 003634 005237 000304 TST31: INC @#TESTN
1052 003640 012706 021774 MOV #BUFF,%6 ;STACK POINTER SETUP
1053 003644 012737 003654 000014 MOV #RETC4,BPTVEC ;RETURN FROM TRAP POINTER
1054 003652 000003 BPT ;TRAP ON THIS INSTRUCTION
1055 003654 022737 003654 021770 RETC4: CMP #.,BUFF-4 ;CHECK FOR INCREMENTED P.C.
1056 003662 001405 BEQ 1$
1057
1058 003664 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1059 003666 012713 000041 MOV #41,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1060 003672 000000 HLT ;WRONG "OLD" PC SAVED ON STACK
1061 ;ON "BPT" TRAP
1062 003674 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
1063 ;"JMP @RO" (000110) TO GET A
1064 ;SCOPE LOOP AND HIT CONTINUE
1065 003676 010700 1$: SCOPE
1066
  
```

K02

MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 23
 DOKDBA.P11 17-FEB-77 08:26

T32 TEST THAT "OLD" CC AND PRIORITY ARE PLACED ON STACK ON A "BPT" TRAP

```

1067 ;*****
1068 ;*TEST 32 TEST THAT "OLD" CC AND PRIORITY ARE PLACED ON STACK ON A "BPT" TRAP
1069 ;*****
1070 003700 005237 000304 TST32: INC 2*STESTN
1071 003704 012706 021774 MOV #BUFF,%6 ;SET UP
1072 003710 012737 003724 000014 MOV #RETD4,BPTVEC ;LOAD VECTOR
1073 003716 005037 177776 CLR CC ;CLEAR CC AND PRIORITY
1074 003722 000003 BPT
1075 003724 013700 021772 RETD4: MOV BUFF-2,%0 ;GET SAVED STATUS OFF STACK
1076 003730 001405 BEQ 1$ ;BRANCH IF ALL 0'S
1077
1078 003732 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1079 003734 012713 000042 MOV #42,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1080 003740 000000 HLT ;WRONG "OLD" PSW SAVED ON STACK
1081 ;ON "BPT" TRAP, EXPECT 0
1082 003742 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
1083 ;"JMP 2RO" (000110) TO GET A
1084 ;SCOPE LOOP AND HIT CONTINUE
1085 003744 010700 1$: SCOPE
1086
1087 003746 012706 021774 MOV #BUFF,%6 ;SET UP
1088 003752 012737 003770 000014 MOV #RETE4,BPTVEC ;SET UP
1089 003760 012737 000357 177776 MOV #357,CC ;PRE SET STATUS
1090 003766 000003 BPT
1091 003770 013700 021772 RETE4: MOV BUFF-2,%0 ;GET THE SAVED STATUS
1092 003774 022700 000357 CMP #357,%0 ;IS IT CORRECT?
1093 004000 001405 BEQ 1$ ;BRANCH IF CORRECT
1094
1095 004002 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1096 004004 012713 000043 MOV #43,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1097 004010 000000 HLT ;WRONG "OLD" PSW SAVED ON STACK
1098 ;ON "BPT" TRAP, EXPECT 357
1099 004012 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
1100 ;"JMP 2RO" (000110) TO GET A
1101 ;SCOPE LOOP AND HIT CONTINUE
1102 004014 010700 1$: SCOPE
1103
1104 ;*****
1105 ;*TEST 33 TEST THAT "NEW" STATUS IS CORRECT ON A "BPT" TRAP
1106 ;*****
1107 004016 005237 000304 TST33: INC 2*STESTN
1108 004022 012706 021774 MOV #BUFF,%6
1109 004026 012737 004050 000014 MOV #RETF4,BPTVEC
1110 004034 005037 000016 CLR BPTVEC+2 ;CLEAR FUTURE PRIORITY AND CC
1111 004040 012737 000357 177776 MOV #357,CC ;PRE SET STATUS
1112 004046 000003 BPT
1113 004050 013700 177776 RETF4: MOV CC,%0 ;GET & TEST 'NEW' STATUS
1114 004054 001405 BEQ 1$ ;BRANCH IF ALL 0'S
1115
1116 004056 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1117 004060 012713 000044 MOV #44,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1118 004064 000000 HLT ;WRONG "NEW" PSW LOADED ON
1119 ;"BPT" TRAP, EXPECT 0
1120 004066 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
1121 ;"JMP 2RO" (000110) TO GET A
1122 ;SCOPE LOOP AND HIT CONTINUE
  
```


M02

T33 TEST THAT "NEW" STATUS IS CORRECT ON A "BPT" TRAP

```

1148
1149
1150 ;*****
1151 ;*TEST 34 TEST THAT A TRAP OCCURS ON AN ILLEGAL INSTRUCTION - JMP RO
1152 ;*****
1152 004170 005237 000304 TST34: INC @STESTN
1153 004174 012706 021774 MOV #BUFF,%6 ;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 %0 ;ILLEGAL INSTRUCTION, SHOULD TRAP
1157
1158 004214 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1159 004216 012713 000046 MOV #46,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1160 004222 000000 HLT ;ILLEGAL INSTR. DID NOT TRAP
1161 004224 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
1162 ;"JMP @RO" (000110) TO GET A
1163 ;SCOPE LOOP AND HIT CONTINUE
1164 004226 010700 RETAS: SCOPE
1165
1166 ;*****
1167 ;*TEST 35 TEST DECREMENT OF STACK POINTER ON AN ILLEGAL INSTRUCTION TRAP
1168 ;*****
1169 004230 005237 000304 TST35: INC @STESTN
1170 004234 012706 021774 MOV #BUFF,%6 ;STACK POINTER SETUP
1171 004240 012737 004250 000004 MOV #RETB5,ERRVEC ;RETURN POINTER
Z 1172 004246 000100 JMP %0 ;RESERVED INSTRUCTION
1173 004250 020627 021770 RETB5: CMP %6,#BUFF-4 ;TEST DECREMENT OF %6
1174 004254 001405 BEQ 1$
1175
1176 004256 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1177 004260 012713 000047 MOV #47,(R3) ;MOVE TO MAIL BOX ($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 @RO" (000110) TO GET A
1182 ;SCOPE LOOP AND HIT CONTINUE
1183 004270 010700 1$: SCOPE
1184
1185 ;*****
1186 ;*TEST 36 TEST THAT PROPER P.C. IS SAVED ON AN ILLEGAL INSTRUCTION TRAP
1187 ;*****
1188 004272 005237 000304 TST36: INC @STESTN
1189 004276 012706 021774 MOV #BUFF,%6 ;STACK POINTER SETUP
1190 004302 012737 004312 000004 MOV #RETC5,ERRVEC ;RETURN FROM TRAP POINTER
Z 1191 004310 000100 JMP %0 ;TRAP ON THIS INSTRUCTION
1192 004312 022737 004312 021770 RETC5: CMP #.,BUFF-4 ;CHECK FOR INCREMENTED P.C.
1193 004320 001405 BEQ 1$
1194
1195 004322 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1196 004324 012713 000050 MOV #50,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1197 004330 000000 HLT ;WRONG "OLD" PC SAVED ON STACK
1198 ;ON ILLEGAL INSTR. TRAP
1199 004332 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
1200 ;"JMP @RO" (000110) TO GET A
1201 ;SCOPE LOOP AND HIT CONTINUE
1202 004334 010700 1$: SCOPE
1203

```

N02

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 26
 DOKDBA.P11 17-FEB-77 08:26

T37 TEST THAT "OLD" CC AND PRIORITY ARE SAVED ON AN ILLEGAL INSTR. TRAP

```

1204 ;*****
1205 ;*TEST 37 TEST THAT "OLD" CC AND PRIORITY ARE SAVED ON AN ILLEGAL INSTR. TRAP
1206 ;*****
1207 004336 005237 000304 TST37: INC 2*STESTN
1208 004342 012706 021774 MOV #BUFF,%6 ;SET UP
1209 004346 012737 004362 000004 MOV #RETDS,ERRVEC ;SET UP
1210 004354 005037 177776 CLR CC ;CLEAR CC AND PRIORITY
1211 Z 004360 000100 JMP %0 ;TRAP
1212 004362 013700 021772 RETDS: MOV BUFF-2,%0 ;GET THE SAVED STATUS
1213 004366 001405 BEQ IS ;BRANCH IF ALL 0'S
1214
1215 004370 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1216 004372 012713 000051 MOV #51,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1217 004376 000000 HLT ;ERROR! SAVED PSW IS INCORRECT
1218 ;ON ILLEGAL INSTR. TRAP, EXPECT 0
1219 004400 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
1220 ;"JMP 2R0" (000110) TO GET A
1221 ;SCOPE LOOP AND HIT CONTINUE
1222 004402 010700 IS: SCOPE
1223
1224 004404 012706 021774 MOV #BUFF,%6 ;SET UP
1225 004410 012737 004426 000004 MOV #RETES,ERRVEC ;SET UP
1226 004416 012737 000357 177776 MOV #357,CC ;PRE SET STATUS
1227 Z 004424 000101 JMP %1
1228 004426 013700 021772 RETES: MOV BUFF-2,%0 ;GET THE SAVED STATUS
1229 004432 022700 000357 CMP #357,%0 ;IS IT CORRECT
1230 004436 001405 BEQ IS ;BRANCH IF SAVED STATUS IS CORRECT
1231
1232 004440 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1233 004442 012713 000052 MOV #52,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1234 004446 000000 HLT ;ERROR! SAVED PSW IS INCORRECT
1235 ;ON ILLEGAL INSTR. TRAP, EXPECT 357
1236 004450 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
1237 ;"JMP 2R0" (000110) TO GET A
1238 ;SCOPE LOOP AND HIT CONTINUE
1239 004452 010700 IS: SCOPE
1240
1241 ;*****
1242 ;*TEST 40 TEST THAT "NEW" STATUS IS CORRECT ON AN ILLEGAL INSTR. TRAP
1243 ;*****
1244 004454 005237 000304 TST40: INC 2*STESTN
1245 004460 012706 021774 MOV #BUFF,%6
1246 004464 012737 004506 000004 MOV #RETF5,ERRVEC
1247 004472 005037 000006 CLR ERRVEC+2 ;CLEAR FUTURE PRIORITY AND CC
1248 004476 012737 000357 177776 MOV #357,CC ;PRE SET STATUS
1249 Z 004504 000100 JMP %0
1250 004506 013700 177776 RETFS: MOV CC,%0 ;GET & TEST 'NEW' STATUS
1251 004512 001405 BEQ IS
1252
1253 004514 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1254 004516 012713 000053 MOV #53,(R3) ;MOVE TO MAIL BOX ($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 2R0" (000110) TO GET A
1259 ;SCOPE LOOP AND HIT CONTINUE
  
```

B03

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 27
 DOKDBA.P11 17-FEB-77 08:26 T40 TEST THAT "NEW" STATUS IS CORRECT ON AN ILLEGAL INSTR. TRAP

1260	004526	005037	177776	15:	CLR	CC	
1261	004532	010700			SCOPE		
1262							
1263	004534	012706	021774		MOV	#BUFF,%6	
1264	004540	012737	004562	000004	MOV	#RETGS,ERRVEC	
1265	004546	012737	000357	000006	MOV	#357,ERRVEC+2	;LOAD 'NEW' STATUS
1266	004554	005037	177776		CLR	CC	;PRE SET STATUS
Z 1267	004560	000100			JMP	%0	
1268	004562	013700	177776	RETGS:	MOV	CC,%0	;GET THE 'NEW' STATUS
1269	004566	022700	000357		CMP	#357,%0	;IS IT CORRECT
1270	004572	001405			BEQ	15	;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 @R0" (000110) TO GET A
1278							; SCOPE LOOP AND HIT CONTINUE
1279	004606	005037	177776	15:	CLR	CC	
1280	004612	010700			SCOPE		

T40 TEST THAT "NEW" STATUS IS CORRECT ON AN ILLEGAL INSTR. TRAP

```

1281
1282
1283
1284
1285 004614 005237 000304 TST41: INC @#STESTN
1286 004620 012706 021774 MOV @BUFF,%6 ;STACK POINTER SETUP
1287 004624 012737 004652 000004 MOV @RETH5,ERRVEC ;RETURN LOCATION
1288 004632 005037 000006 CLR ERRVEC+2
Z 1289 004636 004000 JSR %0,%0 ;ILLEGAL INSTRUCTION (SHOULD TRAP)
1290
1291 004640 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1292 004642 012713 000055 MOV #55,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1293 004646 000000 HLT ;ILLEGAL INSTRUCTION FAILED TO TRAP
1294 004650 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
1295 ;"JMP @RO" (000110) TO GET A
1296 ;SCOPE LOOP AND HIT CONTINUE
1297 004652 010700 RETH5: SCOPE
1298
1299 ;TEST DECREMENT OF STACK POINTER ON A TRAP OPERATION
1300 004654 012706 021774 MOV @BUFF,%6 ;STACK POINTER SETUP
1301 004660 012737 004670 000004 MOV @RETJ,ERRVEC ;RETURN POINTER
Z 1302 004666 004000 JSR %0,%0 ;RESERVED INSTRUCTION
1303 004670 020627 021770 RETJ: CMP %6,@BUFF-4 ;TEST DECREMENT OF %6
1304 004674 001405 BEQ 1$
1305
1306 004676 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1307 004700 012713 000056 MOV #56,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1308 004704 000000 HLT ;R6 DID NOT DECREM BY 2 WORDS
1309 ;ON ILLEGAL INSTR. TRAP
1310 004706 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
1311 ;"JMP @RO" (000110) TO GET A
1312 ;SCOPE LOOP AND HIT CONTINUE
1313 004710 010700 1$: SCOPE
1314
1315
1316
1317
1318 004712 005237 000304 TST42: INC @#STESTN
1319 004716 012706 021774 MOV @BUFF,%6 ;STACK POINTER SETUP
1320 004722 012737 004732 000004 MOV @RETK,ERRVEC ;RETURN FROM TRAP POINTER
Z 1321 004730 004000 INSTK: JSR %0,%0 ;TRAP ON THIS INSTRUCTION
1322 004732 022737 004732 021770 RETK: CMP @INSTK+2,BUFF-4 ;CHECK FOR INCREMENTED P.C.
1323 004740 001405 BEQ 1$
1324
1325 004742 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1326 004744 012713 000057 MOV #57,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1327 004750 000000 HLT ;WRONG "OLD" PC SAVED ON STACK
1328 ;ON ILLEGAL INSTR. TRAP
1329 004752 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
1330 ;"JMP @RO" (000110) TO GET A
1331 ;SCOPE LOOP AND HIT CONTINUE
1332 004754 010700 1$: SCOPE
1333
1334
1335
1336

```

:TEST 43 TEST THAT "OLD" CC AND PRIORITY ARE SAVED ON AN ILLEGAL INSTR. TRAP

T43 TEST THAT "OLD" CC AND PRIORITY ARE SAVED ON AN ILLEGAL INSTR. TRAP

```

1337 004756 005237 000304      TST43:  INC  @#STESTN
1338 004762 012706 021774      MOV  #BUFF,%6      ;SET UP
1339 004766 012737 005002 000004  MOV  #RETL,ERRVEC  ;SET UP
1340 004774 005037 177776      CLR  CC            ;CLEAR CC AND PRIORITY
Z 1341 005000 004000      JSR  %0,%0
1342 005002 013700 021772      RETL:  MOV  BUFF-2,%0  ;GET & TEST SAVED STATUS
1343 005006 001405      BEQ  IS            ;BRANCH IF ALL 0'S
1344
1345 005010 005212      INC  (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1346 005012 012713 000060      MOV  #60,(R3)     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1347 005016 000000      HLT
1348
1349 005020 000240      LOOP
1350
1351
1352 005022 010700      IS:   SCOPE
1353
1354 005024 012706 021774      MOV  #BUFF,%6      ;SET UP
1355 005030 012737 005046 000004  MOV  #RETM,ERRVEC  ;SET UP
1356 005036 012737 000357 177776  MOV  #357,CC       ;PRE SET STATUS
Z 1357 005044 004000      JSR  %0,%0
1358 005046 013700 021772      RETM:  MOV  BUFF-2,%0  ;GET SAVED STATUS
1359 005052 022700 000357      CMP  #357,%0      ;IS IT CORRECT
1360 005056 001405      BEQ  IS
1361
1362 005060 005212      INC  (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1363 005062 012713 000061      MOV  #61,(R3)     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1364 005066 000000      HLT
1365
1366 005070 000240      LOOP
1367
1368
1369 005072 010700      IS:   SCOPE
1370
1371
1372
1373
1374 005074 005237 000304      TST44:  INC  @#STESTN
1375 005100 012706 021774      MOV  #BUFF,%6
1376 005104 012737 005126 000004  MOV  #RETN,ERRVEC
1377 005112 005037 000006      CLR  ERRVEC+2     ;CLEAR FUTURE PRIORITY AND CC
1378 005116 012737 000357 177776  MOV  #357,CC      ;PRE SET STATUS
Z 1379 005124 004000      JSR  %0,%0
1380 005126 013700 177776      RETN:  MOV  CC,%0    ;GET & TEST 'NEW' STATUS
1381 005132 001405      BEQ  IS            ;BRANCH IF ALL 0'S
1382
1383 005134 005212      INC  (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1384 005136 012713 000062      MOV  #62,(R3)     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1385 005142 000000      HLT
1386
1387 005144 000240      LOOP
1388
1389
1390 005146 005037 177776      IS:   CLR  CC
1391 005152 010700      SCOPE
1392

```

```

;*****
;*TEST 44 TEST THAT "NEW" STATUS IS CORRECT ON AN ILLEGAL INSTR. TRAP
;*****

```

E03

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 30
 DOKDBA.P11 17-FEB-77 08:26 44 TEST THAT "NEW" STATUS IS CORRECT ON AN ILLEGAL INSTR. TRAP

1393	005154	012706	021774			MOV	#BUFF,%6	
1394	005160	012737	005202	000004		MOV	#RETO,ERRVEC	
1395	005166	012737	000357	000006		MOV	#357,ERRVEC+2	;LOAD 'NEW' STATUS
1396	005174	005037	177776			CLR	CC	;PRE SET STATUS
Z 1397	005200	004000				JSR	%0,%0	
1398	005202	013700	177776		RETO:	MOV	CC,%0	;GET THE 'NEW' STATUS
1399	005206	022700	000357			CMP	#357,%0	;IS IT CORRECT
1400	005212	001405				BEQ	IS	;BRANCH IF CORRECT
1401								
1402	005214	005212				INC	(R2)	;SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
1403	005216	012713	000063			MOV	#63,(R3)	;MOVE TO MAIL BOX (\$FATAL) THE ERROR
1404	005222	000000				HLT		;ERROR! 'NEW' PSH IS INCORRECT
1405								;ON ILLEGAL INSTR. TRAP
1406	005224	000240				LOOP		;REPLACE THIS INSTRUCTION BY A
1407								; "JMP 2R0" (000110) TO GET A
1408								;SCOPE LOOP AND HIT CONTINUE
1409	005226	005037	177776		IS:	CLR	CC	
1410	005232	010700				SCOPE		
1411								
1412								
1413								;*****
1414								;*TEST 45 TEST THAT A TRAP OCCURES ON AN ILLEGAL ADDRESS (ODD)
1415								;*****
1416	005234	005237	000304		TST45:	INC	2#STESTN	
1417	005240	012706	021774			MOV	#BUFF,%6	;STACK POINTER SETUP
1418	005244	012737	005274	000004		MOV	#RETP,ERRVEC	;RETURN LOCATION
1419	005252	005037	000006			CLR	ERRVEC+2	
1420	005256	005737	000001			TST	1	;ILL.ADRS. (ODD ADDRESS ON WORD INST.)
1421								
1422	005262	005212				INC	(R2)	;SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
1423	005264	012713	000064			MOV	#64,(R3)	;MOVE TO MAIL BOX (\$FATAL) THE ERROR
1424	005270	000000				HLT		;ILLEGAL ADDRESS DID NOT TRAP
1425	005272	000240				LOOP		;REPLACE THIS INSTRUCTION BY A
1426								; "JMP 2R0" (000110) TO GET A
1427								;SCOPE LOOP AND HIT CONTINUE
1428	005274	010700			RETP:	SCOPE		
1429								

F03

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 31
 DOKDBA.P11 17-FEB-77 08:26

T45 TEST THAT A TRAP OCCURES ON AN ILLEGAL ADDRESS (ODD)

```

1430
1431
1432
1433
1434 005276 005237 000304          TST46: INC      2*STESTN
1435 005302 012706 021774          MOV      8*BUFF,%6          ;STACK POINTER SETUP
1436 005306 012737 005320 000004  MOV      8*RETO,ERRVEC      ;RETURN POINTER
1437 005314 005737 000001          TST     1                    ;RESERVED INSTRUCTION
1438 005320 020627 021770          RETQ:  CMP      %6,8*BUFF-4  ;TEST DECREMENT OF %6
1439 005324 001405          BEQ     1$
1440
1441 005326 005212          INC      (R2)                ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1442 005330 012713 000065          MOV      865,(R3)           ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1443 005334 000000          HLT
1444
1445 005336 000240          LOOP
1446
1447
1448 005340 010700          1$:    SCOPE
1449
1450
1451
1452
1453 005342 005237 000304          TST47: INC      2*STESTN
1454 005346 012706 021774          MOV      8*BUFF,%6          ;STACK POINTER SETUP
1455 005352 012737 005364 000004  MOV      8*RETR,ERRVEC      ;RETURN FROM TRAP POINTER
1456 005360 005737 000001          TST     1                    ;TRAP ON THIS INSTRUCTION
1457 005364 022737 005364 021770  RETR:  CMP      8.,BUFF-4    ;CHECK FOR INCREMENTED P.C.
1458 005372 001405          BEQ     1$
1459
1460 005374 005212          INC      (R2)                ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1461 005376 012713 000066          MOV      866,(R3)           ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1462 005402 000000          HLT
1463
1464 005404 000240          LOOP
1465
1466
1467 005406 010700          1$:    SCOPE
1468
1469
1470
1471
1472 005410 005237 000304          TST50: INC      2*STESTN
1473 005414 012706 021774          MOV      8*BUFF,%6          ;SET UP
1474 005420 012737 005436 000004  MOV      8*RETS,ERRVEC      ;SET UP
1475 005426 005037 177776          CLR     CC                  ;CLEAR CC AND PRIORITY
1476 005432 005737 000001          TST     1
1477 005436 013700 021772          RETS:  MOV      8*BUFF-2,%0  ;GET & TEST SAVED STATUS ON STACK
1478 005442 001405          BEQ     1$                  ;BRANCH IF SAVED STATUS IS CORRECT
1479
1480 005444 005212          INC      (R2)                ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1481 005446 012713 000067          MOV      867,(R3)           ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1482 005452 000000          HLT
1483 005454 000240          LOOP
1484
1485

```



```

1486 005456 010700          1S:  SCOPE
1487
1488 005460 012706 021774          MOV  #BUFF,%6          ;SET UP
1489 005464 012737 005504 000004          MOV  #RETT,ERRVEC      ;SET UP
1490 005472 012737 000357 177776          MOV  #357,CC          ;PRE SET STATUS
1491 005500 005737 000001          TST  1
1492 005504 013700 021772          RETT: MOV  BUFF-2,%0      ;GET THE SAVED STATUS OFF STACK
1493 005510 022700 000357          CMP  #357,%0          ;IS IT CORRECT
1494 005514 001405          BEQ  1S               ;BRANCH IF CORRECT
1495
1496 005516 005212          INC  (R2)              ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1497 005520 012713 000070          MOV  #70,(R3)         ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1498 005524 000000          HLT
1499 005526 000240          LOOP                  ;ERROR! SAVED PSW ON STACK IS INCORRECT
1500                                     ;REPLACE THIS INSTRUCTION BY A
1501                                     ;"JMP 2RO" (000110) TO GET A
1502 005530 010700          1S:  SCOPE
1503
1504                                     ;*****
1505                                     ;*TEST 51 TEST THAT "NEW" STATUS IS CORRECT ON AN ODD ADDRESS TRAP
1506                                     ;*****
1507 005532 005237 000304          TST51: INC  #STESTN
1508 005536 012706 021774          MOV  #BUFF,%6
1509 005542 012737 005566 000004          MOV  #RETV,ERRVEC
1510 005550 005037 000006          CLR  ERRVEC+2          ;CLEAR FUTURE PRIORITY AND CC
1511 005554 012737 000357 177776          MOV  #357,CC          ;PRE SET STATUS
1512 005562 005737 000001          TST  1
1513 005566 013700 177776          RETU: MOV  CC,%0        ;GET & TEST THE 'NEW' STATUS
1514 005572 001405          BEQ  1S               ;BRANCH IF CORRECT
1515
1516 005574 005212          INC  (R2)              ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1517 005576 012713 000071          MOV  #71,(R3)         ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1518 005602 000000          HLT
1519 005604 000240          LOOP                  ;ERROR! 'NEW' PSW IS INCORRECT
1520                                     ;REPLACE THIS INSTRUCTION BY A
1521                                     ;"JMP 2RO" (000110) TO GET A
1522 005606 005037 177776          1S:  CLR  CC
1523 005612 010700          SCOPE
1524
1525 005614 012706 021774          MOV  #BUFF,%6
1526 005620 012737 005640 000004          MOV  #RETV,ERRVEC
1527 005626 012737 000357 000006          MOV  #357,ERRVEC+2    ;LOAD 'NEW' STATUS
1528 005634 005737 000001          TST  1
1529 005640 013700 177776          RETV: MOV  CC,%0        ;GET THE 'NEW' STATUS
1530 005644 022700 000357          CMP  #357,%0          ;IS IT CORRECT
1531 005650 001405          BEQ  1S               ;BRANCH IF CORRECT
1532
1533 005652 005212          INC  (R2)              ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1534 005654 012713 000072          MOV  #72,(R3)         ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1535 005660 000000          HLT
1536 005662 000240          LOOP                  ;ERROR! 'NEW' STATUS IS INCORRECT
1537                                     ;REPLACE THIS INSTRUCTION BY A
1538                                     ;"JMP 2RO" (000110) TO GET A
1539 005664 005037 177776          1S:  CLR  CC
1540 005670 010700          SCOPE
1541
    
```

H03

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 33
 DOKDBA.P11 17-FEB-77 08:26

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

```

1542 ;*****
1543 ;*TEST 52 TEST THAT AN OODSOURCE INTERMEDIATE ADDRESS CAUSES AN ODD ADDRESS TRAP
1544 ;*****
1545 005672 005237 000304          TST52: INC      @#STESTN
1546 005676 012706 021774          MOV      #BUFF,%6          ;SET UP STACK POINTER
1547 005702 012737 005736 000004  MOV      #OAERO,4          ;LOAD ERROR VECTOR
1548 005710 005037 000006          CLR      6
1549 005714 012701 000001          MOV      #1,%1            ;LOAD INDEX REGISTER
1550 005720 067100 000000          ADD      @0(1),%0          ;SRC ADRS [0(1)] IS ODD
1551
1552          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1553 005724 005212          MOV      #73,(R3)          ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1554 005726 012713 000073          HLT
1555 005732 000000          LOOP          ;ERROR! ODD ADRES ERROR FAILED TO TRAP
1556 005734 000240          ;REPLACE THIS INSTRUCTION BY A
1557          ;"JMP @RO" (000110) TO GET A
1558          ;SCOPE LOOP AND HIT CONTINUE
1558 005736 010700          OAERO:  SCOPE
1559
1560 005740 012706 021774          MOV      #BUFF,%6
1561 005744 012737 005774 000004  MOV      #OAER1,4
1562 005752 012701 021773          MOV      #BUFF-1,%1
1563 005756 147100 000000          BICB    @0(1),%0          ;SRC INT ADRS [0(1)] IS ODD
1564
1565          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1566 005762 005212          MOV      #74,(R3)          ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1567 005764 012713 000074          HLT
1568 005770 000000          LOOP          ;ERROR! ODD ADRS IN DST FAILED TO TRAP
1569 005772 000240          ;REPLACE THIS INSTRUCTION BY A
1570          ;"JMP @RO" (000110) TO GET A
1571          ;SCOPE LOOP AND HIT CONTINUE
1571 005774 010700          OAER1:  SCOPE
1572
1573 ;*****
1574 ;*TEST 53 TEST THAT AN ODD SOURCE FINAL ADDRESS WILL CAUSE AN ODD ADDRESS TRAP
1575 ;*****
1576 005776 005237 000304          TST53: INC      @#STESTN
1577 006002 012706 021774          MOV      #BUFF,%6
1578 006006 012737 006042 000004  MOV      #OAER2,4
1579 006014 012737 000001 000100  MOV      #1,100
1580 006022 005001          CLR      %1
1581 006024 017100 000100          MOV      @100(1),%0          ;SRC FINAL ADRS IS ODD
1582
1583          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1584 006030 005212          MOV      #75,(R3)          ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1585 006032 012713 000075          HLT
1586 006036 000000          LOOP          ;ERROR! ODD FINAL SRC ADRS FAILED TO TRAP
1587 006040 000240          ;REPLACE THIS INSTRUCTION BY A
1588          ;"JMP @RO" (000110) TO GET A
1589          ;SCOPE LOOP AND HIT CONTINUE
1589 006042 010700          OAER2:  SCOPE
1590
1591 ;*****
1592 ;*TEST 54 TEST THAT AN ODD DEST INTERMEDIATE ADDRESS CAUSES AN ODD ADRS TRAP
1593 ;*****
1594 006044 005237 000304          TST54: INC      @#STESTN
1595 006050 012706 021774          MOV      #BUFF,%6
1596 006054 012737 006104 000004  MOV      #OAER3,4
1597 006062 012701 000001          MOV      #1,%1
  
```

MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 34
 DOKDBA.P11 17-FEB-77 08:26

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

```

1598 006066 074071 000000          XOR      %0,%0(1)          ;DST INT ADRS [0(1)] IS ODD
1599
1600 006072 005212          INC      (R2)              ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1601 006074 012713 000076          MOV      #76,(R3)          ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1602 006100 000000          HLT
1603 006102 000240          LOOP
1604
1605
1606 006104 010700          OAER3:  SCOPE
1607
1608 006106 012706 021774          MOV      #BUFF,%6
1609 006112 012737 006136 000004          MOV      #OAER4,4
1610 006120 005001          CLR      %1
1611 006122 122131          CMPB    (1)+,%(1)+        ;DST INT ADRS IS ODD [(R1) =1]
1612
1613 006124 005212          INC      (R2)              ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1614 006126 012713 000077          MOV      #77,(R3)          ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1615 006132 000000          HLT
1616 006134 000240          LOOP
1617
1618
1619 006136 010700          OAER4:  SCOPE
1620
1621
1622
1623
1624 006140 005237 000304          TST55: INC      #STESTN
1625 006144 012706 021774          MOV      #BUFF,%6
1626 006150 012737 006204 000004          MOV      #OAERS,4
1627 006156 012737 000001 000100          MOV      #1,100
1628 006164 005001          CLR      %1
1629 006166 006771 000100          SXT      @100(1)          ;DST FINAL ADRS IS ODD
1630
1631 006172 005212          INC      (R2)              ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1632 006174 012713 000100          MOV      #100,(R3)         ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1633 006200 000000          HLT
1634 006202 000240          LOOP
1635
1636
1637 006204 010700          OAERS:  SCOPE
1638
1639
1640
1641
1642 006206 005237 000304          TST56: INC      #STESTN
1643 006212 012706 021774          MOV      #BUFF,%6          ;INITIALIZE THE STACK POINTER
1644 006216 012737 006324 000004          MOV      #OAER6,4
1645 C 34 005001          CLR      %1
1646 C 36 012737 021415 000000          MOV      #TEMP+1,0
1647 C 34 012737 021416 000002          MOV      #TEMP+2,2
1648 C 42 012737 123000 021414          MOV      #123000,TEMP
1649 C 50 012737 177246 021416          MOV      #177246,TEMP+2
1650 C 56 123131          CMPB    @(1)+,%(1)+        ;COMP. LOC. (BYTE) TEMP+1 & TEMP+2
1651 006260 001405          BEQ     15
1652
1653 006262 005212          INC      (R2)              ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
  
```

J03

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 35
 DOKDBA.P11 17-FEB-77 08:26

T56 TEST THAT CMPB 2(R)+,2(R)+ DOES NOT FAIL

```

1654 006264 012713 000101      MOV      #101,(R3)      ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1655 006270 000000      HLT
1656 006272 000240      LOOP
1657
1658
1659 006274 022701 000004      15:     CMP      #4,%1
1660 006300 001405      BEQ      25
1661
1662 006302 005212      INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1663 006304 012713 000102      MOV      #102,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1664 006310 000000      HLT
1665 006312 000240      LOOP
1666
1667
1668 006314 000405      25:     BR       OAER8
1669
1670 006316 005212      INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1671 006320 012713 000103      MOV      #103,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1672 006324 000000      OAER6:  HLT
1673 006326 000240      LOOP
1674
1675
1676 006330 010700      OAER8:  SCOPE
1677
1678
1679
1680
1681 006332 005237 000304      TST57:  INC      @#STESTN
1682 006336 012706 021774      MOV      #BUFF,%6
1683 006342 012737 006366 000004      MOV      #OAER7,4
1684 006350 000337 021415      SWAB     TEMP+1      ;DO SWAB USING AN ODD ADDRESS
1685
1686 006354 005212      INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1687 006356 012713 000104      MOV      #104,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1688 006362 000000      HLT
1689 006364 000240      LOOP
1690
1691
1692 006366      OAER7:
1693
1694 006366 012737 000006 000004      MOV      #ERRVEC+2,ERRVEC;RESTORE ODD ADDRESS ERROR TO
1695 006374 005037 000006      CLR      ERRVEC+2    ;HALT AT ERRVEC+2
1696 006400 010700
1697

```

```

;*****
;*TEST 57 TEST THAT SWAB ODD ADDRESS CAUSES AN ODD ADDRESS TRAP
;*****

```

K03

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 36
 DOKDBA.P11 17-FEB-77 08:26

T57 TEST THAT SWAB ODD ADDRESS CAUSES AN ODD ADDRESS TRAP

```

1698
1699
1700
1701
1702 006402 005237 000304          TST60: INC      @#STESTN
1703 006406 012706 021774          MOV      #BUFF,%6
1704 006412 005001          CLR      R1
1705 006414 012737 006454 000014    MOV      #RETAT,TRTVEC ;SET UP TO TRAP TO 14
1706 006422 005037 000016          CLR      TRTVEC+2
1707 006426 012746 000020          MOV      #20,-(6)      ;PUSH 'T' BIT ON THE STACK
1708 006432 012746 006440          MOV      #.+6,-(6)    ;PUSH PC ON THE STACK
1709 006436 000002          RTI      ;SET 'T' BIT
1710 006440 005201          INC      R1           ;TRAP HERE
1711
1712 006442 005212          INC      (R2)        ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1713 006444 012713 000105    MOV      #105,(R3)   ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1714 006450 000000          HLT
1715 006452 000240          LOOP
1716
1717
1718 006454 005701          RETAT:  TST      R1
1719 006456 001405          BEQ      1$
1720
1721 006460 005212          INC      (R2)        ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1722 006462 012713 000106    MOV      #106,(R3)   ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1723 006466 000000          HLT
1724
1725
1726 006470 000240          LOOP
1727
1728
1729 006472 010700          1$:      SCOPE
1730
1731
1732
1733 006474 005237 000304          TST61:  INC      @#STESTN
1734 006500 012706 021774          MOV      #BUFF,%6
1735 006504 012737 006524 000014    MOV      #RETBT,TRTVEC
1736 006512 012746 000020          MOV      #20,-(6)      ;PUSH 'T' BIT ON THE STACK
1737 006516 012746 006524          MOV      #.+6,-(6)    ;PUSH PC ON THE STACK
1738 006522 000002          RTI      ;SET 'T' BIT
1739 006524 020627 021770          RETBT:  CMP      %6,#BUFF-4
1740 006530 001405          BEQ      1$
1741
1742 006532 005212          INC      (R2)        ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1743 006534 012713 000107    MOV      #107,(R3)   ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1744 006540 000000          HLT
1745 006542 000240          LOOP
1746
1747
1748 006544 010700          1$:      SCOPE
1749
1750
1751
1752 006546 005237 000304          TST62:  INC      @#STESTN
1753 006552 012706 021774          MOV      #BUFF,%6
  
```


M03

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 38
 DOKDBA.P11 17-FEB-77 08:26

T63 TEST FOR PROPER PC ON THE STACK, ON A T BIT TRAP (RTT)

1810	006724	022737	006712	021770	RETCT1: CMP	#RETCT2,BUFF-4	;PROPER PC ON THE STACK
1811	006732	001405			BEQ	1\$	
1812							
1813	006734	005212			INC	(R2)	;SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
1814	006736	012713	000114		MOV	#114,(R3)	;MOVE TO MAIL BOX (\$FATAL) THE ERROR
1815	006742	000000			HLT		;ERROR! IMPROPER PC ON THE STACK
1816	006744	000240			LOOP		;REPLACE THIS INSTRUCTION BY A
1817							;"JMP @RO" (000110) TO GET A
1818							;SCOPE LOOP AND HIT CONTINUE
1819	006746	005701			1\$: TST	R1	;DID T BIT ALLOW ONE INSTR
1820							;AFTER RTT BEFORE TRAPPING?
1821	006750	001005			BNE	2\$;YES
1822							
1823	006752	005212			INC	(R2)	;SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
1824	006754	012713	000115		MOV	#115,(R3)	;MOVE TO MAIL BOX (\$FATAL) THE ERROR
1825	006760	000000			HLT		;T BIT DID NOT ALLOW ONE
1826							;INSTR AFTER RTT BEFORE
1827							;TRAPPING
1828	006762	000240			LOOP		;REPLACE THIS INSTRUCTION BY A
1829							;"JMP @RO" (000110) TO GET A
1830							;SCOPE LOOP AND HIT CONTINUE
1831	006764				2\$:		
1832							
1833							
1834	006764	012737	000016	000014	MOV	#16,TRTVEC	;RESTORE 'T' TRAP
1835	006772	005037	000016		CLR	TRTVEC+2	;TO HALT AT 16
1836	006776	010700			SCOPE		

N03

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 39
 DOKDBA.P11 17-FEB-77 08:26

T63 TEST FOR PROPER PC ON THE STACK, ON A T BIT TRAP (RTT)

```

1837
1838
1839 ;*****
1840 ;*TEST 64 TEST NO STACK OVERFLOW TRAP ON A DATI TO STACK LOC. LESS THAN 400
1841 ;*****
1841 007000 005237 000304 TST64: INC 2*STESTN
1842 007004 012706 000376 MOV #376,%6 ;SET STACK POINTER LESS THAN 400
1843 007010 012737 007034 000004 MOV #TDEC1,ERRVEC ;LOAD TRAP VECTOR
1844 007016 005037 000006 CLR ERRVEC+2
1845 007022 005716 TST (6) ;DATI FROM LOC 376
1846 007024 000405 BR TDEC1A
1847
1848 007026 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1849 007030 012713 000116 TDEC1: MOV #116,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1850 007034 000000 HLT ;ERROR! STACK OVERFLOW TRAP OCCURED
1851 ;ON A DATI TO LOC LESS THAN 400
1852 007036 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
1853 ;"JMP JRO" (000110) TO GET A
1854 ;SCOPE LOOP AND HIT CONTINUE
1855 007040 010700 TDEC1A: SCOPE
1856
1857 ;*****
1858 ;*TEST 65 TEST STACK OVERFLOW TRAP ON A DATIP/DATO TO A STACK LOC. LESS THAN 400
1859 ;*****
1860 007042 005237 000304 TST65: INC 2*STESTN
1861 007046 012706 000376 MOV #376,%5
1862 007052 012737 007076 000004 MOV #TDEC2,ERRVEC
1863 007060 005716 CLR (6) ;DATIP/DATO TO BE 376
1864 007062 000240 NOP
1865
1866 007064 005212 INC (R2) ;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 ;ERROR! NO STACK OVERFLOW TRAP (YELLOW)
1869 007074 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
1870 ;"JMP JRO" (000110) TO GET A
1871 ;SCOPE LOOP AND HIT CONTINUE
1872 007076 010700 TDEC2: SCOPE
1873
1874 ;*****
1875 ;*TEST 66 TEST THAT A DATIP/DATOB CAUSES AN OVERFLOW TRAP
1876 ;*****
1877 007100 005237 000304 TST66: INC 2*STESTN
1878 007104 012706 000376 MOV #376,%6
1879 007110 012737 007134 000004 MOV #TDEC2A,ERRVEC
1880 007116 152716 177777 BISB #-1,(6)
1881
1882 007122 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1883 007124 012713 000120 MOV #120,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1884 007130 000000 HLT ;ERROR! NO STACK OVERFLOW TRAP (YELLOW ZONE)
1885 007132 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
1886 ;"JMP JRO" (000110) TO GET A
1887 ;SCOPE LOOP AND HIT CONTINUE
1888 007134 010700 TDEC2A: SCOPE
1889
1890 ;*****
1891 ;*TEST 67 TEST NO STACK OVERFLOW TRAP ON A DATI (BYTE) TO STACK LOC. LESS THAN 4
1892 ;*****
  
```



```

1893 007136 005237 000304          TST67: INC      @#STESTN
1894 007142 012705 001000          MOV      #1000,%5
1895 007146 012706 000376          MOV      #376,%6
1896 007152 012737 007172 000004    MOV      #TDEC3,ERRVEC
1897 007160 124645          CMPB    -(6),-(5)
1898 007162 000405          BR      TDEC6
1899
1900 007164 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1901 007166 012713 000121    TDEC3: MOV      #121,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1902 007172 000000          HLT                    ;ERROR! STACK OVERFLOW TRAP OCCURED
1903                                     ;ON A DATI TO STACK LOC UNDER 400
1904 007174 000240          LOOP                   ;REPLACE THIS INSTRUCTION BY A
1905                                     ;"JMP 2R0" (000110) TO GET A
1906                                     ;SCOPE LOOP AND HIT CONTINUE
1907 007176 010700    TDEC6: SCOPE
1908
1909 007200 012706 000400          MOV      #400,%6
1910 007204 012737 007224 000004    MOV      #TDEC4,ERRVEC
1911 007212 134546          BITB    -(5),-(6)
1912 007214 000405          BR      TDEC6A
1913
1914 007216 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1915 007220 012713 000122    TDEC4: MOV      #122,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1916 007224 000000          HLT                    ;ERROR! STACK OVERFLOW TRAP OCCURED
1917                                     ;ON A DATI TO STACK LOC UNDER 400
1918 007226 000240          LOOP                   ;REPLACE THIS INSTRUCTION BY A
1919                                     ;"JMP 2R0" (000110) TO GET A
1920                                     ;SCOPE LOOP AND HIT CONTINUE
1921 007230 010700    TDEC6A: SCOPE
1922
1923                                     ;*****
1924                                     ;*TEST 70 TEST THAT OVERFLOW TRAP DOES NOT LOSE INFORMATION (OLD PC & PS)
1925                                     ;*****
1926 007232 005237 000304          TST70: INC      @#STESTN
1927 007236 012706 000400          MOV      #400,%6
1928 007242 005037 000376          CLR     376          ;STATUS WORD OF LOC 10
1929 007246 005037 000374          CLR     374
1930 007252 012737 007304 000004    MOV      #TDEC5,ERRVEC ;RETURN TO LOC 4
1931 007260 012737 000017 177776    MOV      #17,CC      ;PRE SET STATUS
1932 007266 005246          INC     -(6)
1933
1934 007270 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1935 007272 012713 000123    TDEC5A: MOV      #123,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1936 007276 000000          HLT                    ;ERROR! NO STACK OVERFLOW TRAP (YELLOW)
1937 007300 000240          LOOP                   ;REPLACE THIS INSTRUCTION BY A
1938                                     ;"JMP 2R0" (000110) TO GET A
1939                                     ;SCOPE LOOP AND HIT CONTINUE
1940 007302 000415          BR      TDEC5B          ;GO TO SCOPE
1941 007304 022737 000001 000376    TDEC5: CMP      #1,376 ;WAS INC -(6) EXECUTED
1942 007312 001405          BEQ
1943
1944 007314 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1945 007316 012713 000124    TDEC5: MOV      #124,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1946 007322 000000          HLT                    ;ERROR! INSTRUCTION CAUSING YELLOW
1947                                     ;ZONE VIOLATION WAS ABORTED. IT
1948                                     ;SHOULD HAVE BEEN EXECUTED BEFORE
    
```

```

1949                                     ; TRAPPING
1950 007324 000240                       LOOP      ; REPLACE THIS INSTRUCTION BY A
1951                                     ; "JMP 2R0" (000110) TO GET A
1952                                     ; SCOPE LOOP AND HIT CONTINUE
1953 007326 022737 000001 000374 1S:    CMP      #1,374 ; WAS STATUS SAVED NOTE: INC DOES NOT
1954                                     ; AFFECT 'C' BIT IN STATUS.
1955 007334 001400                       BEQ      TDECSB
1956 007336 010700                       TDECSB:  SCOPE
1957
1958                                     ; *****
1959                                     ; *TEST 71 TEST THAT A RESERVED INST CAUSES AN OVERFLOW TRAP
1960                                     ; *****
1961                                     ; *****
1962 007340 005237 000304 000000 000000 TST71:  INC      @#STESTN ; SET UP STACK TO OVERFLOW
1963 007344 012706 000400 000000 000000 MOV      #400,%6 ; SET UP INST VECTOR
1964 007350 012737 007404 000010 000000 MOV      @VDEC2,10 ; SET UP OVERFLOW VECTOR
1965 007356 012737 007410 000004 000000 MOV      @VDEC1,4 ; THIS TRAP SHOULD CAUSE OVERFLOW
1966 007364 075040                       75040
1967
1968 007366 005212                       INC      (R2) ; SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1969 007370 012713 000125 000000 000000 MOV      #125,(R3) ; MOVE TO MAIL BOX ($FATAL) THE ERROR
1970 007374 000000                       HLT      ; NO TRAP OCCURRED
1971
1972 007376 005212                       INC      (R2) ; SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1973 007400 012713 000126 000000 000000 MOV      #126,(R3) ; MOVE TO MAIL BOX ($FATAL) THE ERROR
1974 007404 000000                       HLT      ; TRAP FLAG OVERFLOW DID NOT OCCUR
1975 007406 000240                       LOOP      ; REPLACE THIS INSTRUCTION BY A
1976                                     ; "JMP 2R0" (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 000000 000000 TST72:  INC      @#STESTN ; SET UP STACK TO OVERFLOW
1984 007416 012706 000400 000000 000000 MOV      #400,%6 ; SET UP INST VECTOR
1985 007422 012737 007456 000020 000000 MOV      @VDEC4,20 ; SET UP OVERFLOW VECTOR
1986 007430 012737 007462 000004 000000 MOV      @VDEC3,4 ; THIS TRAP SHOULD CAUSE OVERFLOW
1987 007436 000004                       IOT
1988
1989 007440 005212                       INC      (R2) ; SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1990 007442 012713 000127 000000 000000 MOV      #127,(R3) ; MOVE TO MAIL BOX ($FATAL) THE ERROR
1991 007446 000000                       HLT      ; NO TRAP OCCURRED
1992
1993 007450 005212                       INC      (R2) ; SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1994 007452 012713 000130 000000 000000 MOV      #130,(R3) ; MOVE TO MAIL BOX ($FATAL) THE ERROR
1995 007456 000000                       HLT      ; TRAP FLAG OVERFLOW DID NOT OCCUR
1996 007460 000240                       LOOP      ; REPLACE THIS INSTRUCTION BY A
1997                                     ; "JMP 2R0" (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                                     ; *****
2003 007464 005237 000304 000000 000000 TST73:  INC      @#STESTN ; SET UP STACK TO OVERFLOW
2004 007470 012706 000400 000000 000000 MOV      #400,%6

```



```

2061 007660 010700          VDEC9: SCOPE          ;NORMAL OVERFLOW RETURN
2062
2063
2064
2065          ;*****.*****
2066          ;*TEST 76 TEST THAT AN ILLEGAL INSTR. CAUSES AN OVERFLOW TRAP
2067          ;*****
2066 007662 005237 000304    TST76: INC      2*STESTN
2067 007666 012706 000400    MOV      #400,%6          ;SET UP STACK TO OVERFLOW
2068 007672 012737 007726 000004  MOV      #VDEC12,4       ;SET UP INST VECTOR
2069 007700 012737 007732 000004  MOV      #VDEC11,4       ;SET UP OVERFLOW VECTOR
2070 007706 004700          ILLA          ;THIS TRAP SHOULD CAUSE OVERFLOW
2071
2072 007710 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2073 007712 012713 000137    MOV      #137,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2074 007716 000000          HLT          ;NO TRAP OCCURRED
2075
2076 007720 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2077 007722 012713 000140    MOV      #140,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2078 007726 000000          VDEC12: HLT          ;TRAP FLAG OVERFLOW DID NOT OCCUR
2079 007730 000240          LOOP         ;REPLACE THIS INSTRUCTION BY A
2080          ;"JMP 2R0" (000110) TO GET A
2081          ;SCOPE LOOP AND HIT CONTINUE
2082          ;NORMAL OVERFLOW RETURN
2082 007732 010700          VDEC11: SCOPE
2083 007734 020627 000370    CMP      %6,#370         ;STACK PUSHED FOUR WORDS?
2084 007740 001405          BEQ
2085
2086 007742 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2087 007744 012713 000141    MOV      #141,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2088 007750 000000          HLT          ;TRAP OVERFLOW DID NOT OCCUR
2089 007752 000240          LOOP         ;REPLACE THIS INSTRUCTION BY A
2090          ;"JMP 2R0" (000110) TO GET A
2091          ;SCOPE LOOP AND HIT CONTINUE
2092 007754 010700          IS:   SCOPE
2093
2094
2095          ;*****.*****
2096          ;*TEST 77 TEST THAT AN ILLEGAL INSTR. CAUSES AN OVERFLOW TRAP
2097          ;*****
2097 007756 005237 000304    TST77: INC      2*STESTN
2098 007762 012706 000400    MOV      #400,%6          ;SET UP STACK TO OVERFLOW
2099 007766 012737 010022 000004  MOV      #VDEC14,4       ;SET UP INST VECTOR
2100 007774 012737 010026 000004  MOV      #VDEC13,4       ;SET UP OVERFLOW VECTOR
2101 010002 000100          ILLB          ;THIS TRAP SHOULD CAUSE OVERFLOW
2102
2103 010004 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2104 010006 012713 000142    MOV      #142,(R3)       ;MOVE TO MAIL BOX (%FATAL) THE ERROR
2105 010012 000000          HLT          ;NO TRAP OCCURRED
2106
2107 010014 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2108 010016 012713 000143    MOV      #143,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2109 010022 000000          VDEC14: HLT          ;TRAP FLAG OVERFLOW DID NOT OCCUR
2110 010024 000240          LOOP         ;REPLACE THIS INSTRUCTION BY A
2111          ;"JMP 2R0" (000110) TO GET A
2112          ;SCOPE LOOP AND HIT CONTINUE
2113 010026 010700          VDEC13: SCOPE
2114          ;NORMAL OVERFLOW RETURN
2115
2116          ;INSTRUCTION EQUATE STATEMENTS
    
```

T77 TEST THAT AN ILLEGAL INSTR. CAUSES AN OVERFLOW TRAP

```

2117      :      4510   =JSR   5,(0)
2118      :      005046 =CLR   -(6)
2119      :      010046 =MOV   %0,-(6)
2120      :      006746 =SXT   -(6)
2121      :      074046 =XOR   %0,-(6)
2122
2123      : *****
2124      : *TEST 100 TEST THAT THE INSTRUCTION (4510) CAUSES STACK OVERFLOW CONDITION
2125      : *****
2126      010030 005237 000304      TST100: INC      2*STESTN
2127      010034 012706 000400      MOV      #400,%6 ;SET STACK POINTER
2128      010040 005000      CLR      %0      ;PRE SET R0
2129      010042 012737 010064 000004      MOV      #VDEC15,4 ;LOAD ERROR VECTOR
2130      010050 004510      4510      ;CAUSE OVERFLOW
2131
2132      010052 005212      INC      (R2)      ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2133      010054 012713 000144      MOV      #144,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2134      010060 000000      HLT      ;ERROR! OVERFLOW FAILED TO TRAP
2135      010062 000240      LOOP     ;REPLACE THIS INSTRUCTION BY 15
2136      : "JMP 2R0" (000110) TO GET 15
2137      : SCOPE LOOP AND HIT CONTINUE
2138      010064 022706 000372      VDEC15: CMP      #372,%6 ;HAS STACK POINTER MOVED BY 6
2139      010070 001405      BEQ      1$      ;(2 FOR THE AUTO DECREMENT + 4 FOR
2140
2141      010072 005212      INC      (R2)      ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2142      010074 012713 000145      MOV      #145,(R3) ;MOVE TO MAIL BOX ($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 000240      LOOP     ;REPLACE THIS INSTRUCTION BY 15
2148      : "JMP 2R0" (000110) TO GET 15
2149      : SCOPE LOOP AND HIT CONTINUE
2150      010104 010700      1$:      SCOPE
2151
2152      : *****
2153      : *TEST 101 TEST THAT THE INSTRUCTION (005046) CAUSES STACK OVERFLOW CONDITION
2154      : *****
2155      010106 005237 000304      TST101: INC      2*STESTN
2156      010112 012706 000400      MOV      #400,%6 ;SET STACK POINTER
2157      010116 005000      CLR      %0      ;PRE SET R0
2158      010120 012737 010142 000004      MOV      #VDEC16,4 ;LOAD ERROR VECTOR
2159      010126 005046      005046      ;CAUSE OVERFLOW
2160
2161      010130 005212      INC      (R2)      ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2162      010132 012713 000146      MOV      #146,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2163      010136 000000      HLT      ;ERROR! OVERFLOW FAILED TO TRAP
2164      010140 000240      LOOP     ;REPLACE THIS INSTRUCTION BY 16
2165      : "JMP 2R0" (000110) TO GET 16
2166      : SCOPE LOOP AND HIT CONTINUE
2167      010142 022706 000372      VDEC16: CMP      #372,%6 ;HAS STACK POINTER MOVED BY 6
2168      010146 001405      BEQ      1$      ;(2 FOR THE AUTO DECREMENT + 4 FOR
2169
2170      010150 005212      INC      (R2)      ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2171      010152 012713 000147      MOV      #147,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2172      010156 000000      HLT      ;THE ERROR TRAP)

```

T101 TEST THAT THE INSTRUCTION (005046) CAUSES STACK OVERFLOW CONDITION

```

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 16
2177                                     ;"JMP 2RD" (000110) TO GET 16
2178                                     ;SCOPE LOOP AND HIT CONTINUE
2179 010162 010700                       15:  SCOPE
2180
2181                                     ;*****
2182                                     ;*TEST 102 TEST THAT THE INSTRUCTION (010046) CAUSES STACK OVERFLOW CONDITION
2183                                     ;*****
2184 010164 005237 000304                 TST102: INC      2#STESTN
2185 010170 012706 000400                 MOV      #400,%6 ;SET STACK POINTER
2186 010174 005000                       CLR      %0      ;PRE SET R0
2187 010176 012737 010220 000004         MOV      #VDEC17,4 ;LOAD ERROR VECTOR
2188 010204 010046                       010046         ;CAUSE OVERFLOW
2189
2190 010206 005212                       INC      (R2)    ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2191 010210 012713 000150                 MOV      #150,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2192 010214 000000                       HLT
2193 010216 000240                       LOOP          ;ERROR! OVERFLOW FAILED TO TRAP
2194                                     ;REPLACE THIS INSTRUCTION BY 17
2195                                     ;"JMP 2RD" (000110) TO GET 17
2196 010220 022706 000372                 VDEC17: CMP     #372,%6
2197 010224 001405                       BEQ     15      ;SCOPE LOOP AND HIT CONTINUE
2198                                     ;HAS STACK POINTER MOVED BY 6
2199                                     ;(2 FOR THE AUTO DECREMENT + 4 FOR
2200 010226 005212                       INC      (R2)    ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2201 010230 012713 000151                 MOV      #151,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2202 010234 000000                       HLT          ;THE ERROR TRAP)
2203                                     ;R6 DID NOT DECREMENT BY 6 WHEN
2204                                     ;AN INSTRUCTION DOING AUTO-DECREMENT
2205 010236 000240                       LOOP          ;OF R6 CAUSED YELLOW ZONE VIOLATION
2206                                     ;REPLACE THIS INSTRUCTION BY 17
2207                                     ;"JMP 2RD" (000110) TO GET 17
2208 010240 010700                       15:  SCOPE                               ;SCOPE LOOP AND HIT CONTINUE
2209
2210                                     ;*****
2211                                     ;*TEST 103 TEST THAT THE INSTRUCTION (006746) CAUSES STACK OVERFLOW CONDITION
2212                                     ;*****
2213 010242 005237 000304                 TST103: INC      2#STESTN
2214 010246 012706 000400                 MOV      #400,%6 ;SET STACK POINTER
2215 010252 005000                       CLR      %0      ;PRE SET R0
2216 010254 012737 010276 000004         MOV      #VDEC18,4 ;LOAD ERROR VECTOR
2217 010262 006746                       006746         ;CAUSE OVERFLOW
2218
2219 010264 005212                       INC      (R2)    ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2220 010266 012713 000152                 MOV      #152,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2221 010272 000000                       HLT          ;ERROR! OVERFLOW FAILED TO TRAP
2222 010274 000240                       LOOP          ;REPLACE THIS INSTRUCTION BY 18
2223                                     ;"JMP 2RD" (000110) TO GET 18
2224                                     ;SCOPE LOOP AND HIT CONTINUE
2225 010276 022706 000372                 VDEC18: CMP     #372,%6
2226 010302 001405                       BEQ     15      ;HAS STACK POINTER MOVED BY 6
2227                                     ;(2 FOR THE AUTO DECREMENT + 4 FOR
2228 010304 005212                       INC      (R2)    ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
    
```

H04

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 46
 DOKDBA.P11 17-FEB-77 08:26

T103 TEST THAT THE INSTRUCTION (006746) CAUSES STACK OVERFLOW CONDITION

```

2229 010306 012713 000153      MOV      #153,(R3)      ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2230 010312 000000      HLT                      ;THE ERROR TRAP)
2231                                ;R6 DID NOT DECREMENT BY 6 WHEN
2232                                ;AN INSTRUCTION DOING AUTO-DECREMENT
2233                                ;OF R6 CAUSED YELLOW ZONE VIOLATION
2234 010314 000240      LOOP                     ;REPLACE THIS INSTRUCTION BY 18
2235                                ;"JMP 2RO" (000110) TO GET 18
2236                                ;SCOPE LOOP AND HIT CONTINUE
2237 010316 010700      1$:      SCOPE
2238
2239                                ;*****
2240                                ;*TEST 104 TEST THAT THE INSTRUCTION (074046) CAUSES STACK OVERFLOW CONDITION
2241                                ;*****
2242 010320 005237 000304      TST104: INC      @RSTESTN
2243 010324 012706 000400      MOV      #400,%6 ;SET STACK POINTER
2244 010330 005000      CLR      %0              ;PRE SET R0
2245 010332 012737 010354 000004  MOV      #VDEC19,4      ;LOAD ERROR VECTOR
2246 010340 074046      074046      ;CAUSE OVERFLOW
2247
2248 010342 005212      INC      (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2249 010344 012713 000154      MOV      #154,(R3)      ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2250 010350 000000      HLT                      ;ERROR! OVERFLOW FAILED TO TRAP
2251 010352 000240      LOOP                     ;REPLACE THIS INSTRUCTION BY 19
2252                                ;"JMP 2RO" (000110) TO GET 19
2253                                ;SCOPE LOOP AND HIT CONTINUE
2254 010354 022706 000372      VDEC19: CMP      #372,%6      ;HAS STACK POINTER MOVED BY 6
2255 010360 001405      BEQ      1$              ;(2 FOR THE AUTO DECREMENT + 4 FOR
2256
2257 010362 005212      INC      (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2258 010364 012713 000155      MOV      #155,(R3)      ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2259 010370 000000      HLT                      ;THE ERROR TRAP)
2260                                ;R6 DID NOT DECREMENT BY 6 WHEN
2261                                ;AN INSTRUCTION DOING AUTO-DECREMENT
2262                                ;OF R6 CAUSED YELLOW ZONE VIOLATION
2263 010372 000240      LOOP                     ;REPLACE THIS INSTRUCTION BY 19
2264                                ;"JMP 2RO" (000110) TO GET 19
2265                                ;SCOPE LOOP AND HIT CONTINUE
2266 010374 010700      1$:      SCOPE
2267
2268
  
```

T104 TEST THAT THE INSTRUCTION (074046) CAUSES STACK OVERFLOW CONDITION

```

2269
2270
2271
2272
2273 010376 005237 000304
2274
2275
2276 010402 012737 010500 000004
2277 010410 012706 000402
2278 010414 010605
2279 010416 005746
2280 010420 012706 001002
2281 010424 010605
2282 010426 005746
2283 010430 012706 002002
2284 010434 010605
2285 010436 005746
2286 010440 012706 004002
2287 010444 010605
2288 010446 005746
2289 010450 012706 010002
2290 010454 010605
2291 010456 005746
2292 010460 012706 020000
2293 010464 010605
2294 010466 005746
2295 010470 000405
2296
2297 010472 005212
2298 010474 012713 000156
2299 010500 000000
2300
2301
2302
2303 010502 000240
2304
2305
2306 010504 010700
2307
2308
2309
2310
2311 010506 005237 000304
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324

;*****
;TEST 105 TEST FOR FALSE OVERFLOW TRAP
;*****
TST105: INC 2*STESTN
;PROGRAM MAY HAVE RELOADED IF OVERFLOW FAILS

MOV #FOVER,4 ;SET UP OVERFLOW POINTER
MOV #402,%6
MOV R6,R5
TST -(6) ;SHOULD NOT OVERFLOW
MOV #1002,%6
MOV R6,R5
TST -(6) ;SHOULD NOT OVERFLOW
MOV #2002,%6
MOV R6,R5
TST -(6) ;SHOULD NOT OVERFLOW
MOV #4002,%6
MOV R6,R5
TST -(6) ;SHOULD NOT OVERFLOW
MOV #10002,%6
MOV R6,R5
TST -(6)
MOV #20000,%6 ;SHOULD NOT OVERFLOW
MOV R6,R5
TST -(6)
BR FOV1

FOVER: INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
MOV #156,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
HLT ;FALSE STACK OVERFLOW TRAP OCCURRED
;ON DOING AN AUTO-DECREMENT OF R6
;THE INITIAL R6 IS IN SAVED IN R5
;CHECK STACK TO FIND WHERE
;REPLACE THIS INSTRUCTION BY A
;"JMP 2A0" (000110) TO GET A
;SCOPE LOOP AND HIT CONTINUE

FOV1: SCOPE

;*****
;TEST 106 TEST FOR REPEATED TRAPS AND OVERFLOW OF STACK
;*****
TST106: INC 2*STESTN
;WHAT THE STACK LOOKS LIKE WHEN INSTRUCTION AT "04" IS BEING EXECUTED
OVFLW PC=05,STAT=0 04 342 BOTTOM OF STACK
TRT PC=04,STAT=4 4 344
OVFLW PC=05,STAT=0 0 346
IOT PC=03,STAT=3 03A 350
OVFLW PC=05,STAT=0 0 352
IOT PC=03,STAT=3 3 354
OVFLW PC=05,STAT=0 0 356
TRAP PC=02,STAT=2 02A 360
OVFLW PC=05,STAT=0 0 362
TRAP PC=02,STAT=2 2 364
OVFLW PC=05,STAT=0 0 366
TRAP PC=02,STAT=2 0 370

```


T106 TEST FOR REPEATED TRAPS AND OVERFLOW OF STACK

2325					OVFLW	PC=05,STAT=0	01	372	
2326							1	374	
2327					EMT	PC=01,STAT=1	00A	376	
2328							17	400	
2329								402	INITIAL STACK POINTER
2330	010512	012706	000402		MOV	#402,%5			; INITIALIZE STACK POINTER
2331	010516	012737	010632	000030	MOV	#01,EMTVEC			; NEW PC POINTS TO NEXT INSTR.
2332	010524	012737	000001	000032	MOV	#1,EMTVEC+2			; NEW PSW
2333	010532	012737	010646	000034	MOV	#02,TRAPVEC			; NEW PC POINTS TO NEXT INSTR
2334	010540	012737	000002	000036	MOV	#2,TRAPVEC+2			; NEW PSW
2335	010546	012737	010662	000020	MOV	#03,IOTVEC			; NEW PC POINTS TO NEXT INSTR
2336	010554	012737	000003	000022	MOV	#3,IOTVEC+2			; NEW PSW
2337	010562	012737	010676	000014	MOV	#04,TRTVEC			; NEW PC POINTS TO NEXT INSTR
2338	010570	012737	000004	000016	MOV	#4,TRTVEC+2			; NEW PSW
2339	010576	012737	011324	000004	MOV	#05,ERRVEC			; NEW PC POINTS TO NEXT INSTR
2340	010604	005037	000006		CLR	ERRVEC+2			; NEW PSW
2341	010610	012737	000017	177776	MOV	#17,CC			; PRESENT PSW
2342	010616	104000			EMT				
2343	010620						00:		
2344							001:		
2345	010620	005212			INC	(R2)			; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2346	010622	012713	000157		MOV	#157,(R3)			; MOVE TO MAIL BOX (\$FATAL) THE ERROR
2347	010626	000000			HLT		00A:		; EMT NOT EXECUTED
2348	010630	000240			LOOP				; REPLACE THIS INSTRUCTION BY A
2349									; "JMP 2R0" (000110) TO GET A
2350									; SCOPE LOOP AND HIT CONTINUE
2351	010632	104400			TRAP		01:		; RETURN HERE FROM PREVIOUS YELLOW ZONE
2352									; STACK VIOLATION
2353	010634						011:		
2354									
2355	010634	005212			INC	(R2)			; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2356	010636	012713	000160		MOV	#160,(R3)			; MOVE TO MAIL BOX (\$FATAL) THE ERROR
2357	010642	000000			HLT		01A:		; TRAP NOT EXECUTED
2358	010644	000240			LOOP				; REPLACE THIS INSTRUCTION BY A
2359									; "JMP 2R0" (000110) TO GET A
2360									; SCOPE LOOP AND HIT CONTINUE
2361	010646	000004			IOT		02:		; RETURN HERE FROM PREVIOUS YELLOW ZONE
2362									; STACK VIOLATION
2363	010650						021:		
2364									
2365	010650	005212			INC	(R2)			; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2366	010652	012713	000161		MOV	#161,(R3)			; MOVE TO MAIL BOX (\$FATAL) THE ERROR
2367	010656	000000			HLT		02A:		; IOT NOT EXECUTED
2368	010660	000240			LOOP				; REPLACE THIS INSTRUCTION BY A
2369									; "JMP 2R0" (000110) TO GET A
2370									; SCOPE LOOP AND HIT CONTINUE
2371	010662	000003			BPT		03:		; RETURN HERE FROM PREVIOUS YELLOW ZONE
2372									; STACK VIOLATION
2373	010664						031:		
2374									
2375	010664	005212			INC	(R2)			; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2376	010666	012713	000162		MOV	#162,(R3)			; MOVE TO MAIL BOX (\$FATAL) THE ERROR
2377	010672	000000			HLT		03A:		; BPT NOT EXECUTED
2378	010674	000240			LOOP				; REPLACE THIS INSTRUCTION BY A
2379									; "JMP 2R0" (000110) TO GET A
2380									; SCOPE LOOP AND HIT CONTINUE

K04

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 49
 DOKDBA.P11 17-FEB-77 08:26

T106 TEST FOR REPEATED TRAPS AND OVERFLOW OF STACK

2381	010676	022706	000342	04:	CMP	#342,%6	; IS STACK POINTER
2382	010702	001405			BEQ	20\$; POSITIONED PROPERLY
2383							
2384	010704	005212			INC	(R2)	; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2385	010706	012713	000163		MOV	#163,(R3)	; MOVE TO MAIL BOX (\$FATAL) THE ERROR
2386	010712	000000			HLT		; ERROR! INCORRECT STACK POINTER AFTER
2387							; 4 REPETITIVE STACK OVERFLOW TRAPS
2388	010714	000240			LOOP		; REPLACE THIS INSTRUCTION BY A
2389							; "JMP 2R0" (000110) TO GET A
2390							; SCOPE LOOP AND HIT CONTINUE
2391	010716	012700	000402	20\$:	MOV	#402,%0	
2392	010722	022740	000017		CMP	#17,-(0)	; CORRECT OLD PSW SAVED?
2393	010726	001405			BEQ	1\$	
2394							
2395	010730	005212			INC	(R2)	; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2396	010732	012713	000164		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 2R0" (000110) TO GET A
2401							; SCOPE LOOP AND HIT CONTINUE
2402	010742	022740	010620	1\$:	CMP	#001,-(C)	
2403	010746	001405			BEQ	2\$	
2404							
2405	010750	005212			INC	(R2)	; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2406	010752	012713	000165		MOV	#165,(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 2R0" (000110) TO GET A
2411							; SCOPE LOOP AND HIT CONTINUE
2412	010762	022740	000001	2\$:	CMP	#1,-(0)	
2413	010766	001405			BEQ	3\$	
2414							
2415	010770	005212			INC	(R2)	; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2416	010772	012713	000166		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	011000	000240			LOOP		; REPLACE THIS INSTRUCTION BY A
2420							; "JMP 2R0" (000110) TO GET A
2421							; SCOPE LOOP AND HIT CONTINUE
2422	011002	022740	010632	3\$:	CMP	#01,-(0)	
2423	011006	001405			BEQ	4\$	
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 2R0" (000110) TO GET A
2431							; SCOPE LOOP AND HIT CONTINUE
2432	011022	022740	000000	4\$:	CMP	#0,-(0)	
2433	011026	001405			BEQ	5\$	
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

2437	011036	000000			HLT			: CORRECT PSW WAS NOT SAVED ON
2438								: "TRAP" TRAP
2439	011040	000240			LOOP			: REPLACE THIS INSTRUCTION BY A
2440								: "JMP 2R0" (000110) TO GET A
2441								: SCOPE LOOP AND HIT CONTINUE
2442	011042	022740	010634	55:	CMP	#011,-(0)		
2443	011046	001405			BEG	65		
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 2R0" (000110) TO GET A
2451								: SCOPE LOOP AND HIT CONTINUE
2452	011062	022740	000002	65:	CMP	#2,-(0)		
2453	011066	001405			BEG	75		
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 2R0" (000110) TO GET A
2461								: SCOPE LOOP AND HIT CONTINUE
2462	011102	022740	010646	75:	CMP	#02,-(0)		
2463	011106	001405			BEG	85		
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 2R0" (000110) TO GET A
2471								: SCOPE LOOP AND HIT CONTINUE
2472	011122	022740	000000	85:	CMP	#0,-(0)		
2473	011126	001405			BEG	95		
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 2R0" (000110) TO GET A
2481								: SCOPE LOOP AND HIT CONTINUE
2482	011142	022740	010650	95:	CMP	#021,-(0)		
2483	011146	001405			BEG	105		
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 2R0" (000110) TO GET A
2491								: SCOPE LOOP AND HIT CONTINUE
2492	011162	022740	000003	105:	CMP	#3,-(0)		

011166	001405		BEQ	11\$	
011170	005212		INC	(R2)	: SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
011172	012713	000176	MOV	#176, (R3)	: MOVE TO MAIL BOX (\$FATAL) THE ERROR
011176	000000		HLT		: CORRECT PSW WAS NOT SAVED ON STACK
011200	000240		LOOP		: OVERFLOW FOLLOWING IOT
011202	022740	010662	11\$: CMP	#03, -(0)	: REPLACE THIS INSTRUCTION BY A
011206	001405		BEQ	12\$: "JMP 2R0" (000110) TO GET A
011210	005212		INC	(R2)	: SCOPE LOOP AND HIT CONTINUE
011212	012713	000177	MOV	#177, (R3)	: SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
011216	000000		HLT		: MOVE TO MAIL BOX (\$FATAL) THE ERROR
011220	000240		LOOP		: CORRECT PC WAS NOT SAVED ON STACK
011222	022740	000000	12\$: CMP	#0, -(0)	: OVERFLOW FOLLOWING IOT
011226	001405		BEQ	13\$: REPLACE THIS INSTRUCTION BY A
011230	005212		INC	(R2)	: "JMP 2R0" (000110) TO GET A
011232	012713	000200	MOV	#200, (R3)	: SCOPE LOOP AND HIT CONTINUE
011236	000000		HLT		: SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
011240	000240		LOOP		: MOVE TO MAIL BOX (\$FATAL) THE ERROR
011242	022740	010664	13\$: CMP	#031, -(0)	: CORRECT PSW WAS NOT SAVED
011246	001405		BEQ	14\$: ON BPT TRAP
011250	005212		INC	(R2)	: REPLACE THIS INSTRUCTION BY A
011252	012713	000201	MOV	#201, (R3)	: "JMP 2R0" (000110) TO GET A
011256	000000		HLT		: SCOPE LOOP AND HIT CONTINUE
011260	000240		LOOP		: SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
011262	022740	000004	14\$: CMP	#4, -(0)	: MOVE TO MAIL BOX (\$FATAL) THE ERROR
011266	001405		BEQ	15\$: CORRECT PC WAS NOT SAVED
011270	005212		INC	(R2)	: ON BPT TRAP
011272	012713	000202	MOV	#202, (R3)	: REPLACE THIS INSTRUCTION BY A
011276	000000		HLT		: "JMP 2R0" (000110) TO GET A
011300	000240		LOOP		: SCOPE LOOP AND HIT CONTINUE
011302	022740	010676	15\$: CMP	#04, -(0)	: SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
011306	001405		BEQ	07	: MOVE TO MAIL BOX (\$FATAL) THE ERROR
011310	005212		INC	(R2)	: CORRECT PSW WAS NOT SAVED ON STACK
011312	012713	000203	MOV	#203, (R3)	: OVERFLOW FOLLOWING BPT
011316	000000		STKERR: HLT		

N04

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 52
 DOKDBA.P11 17-FEB-77 08:26

T106 TEST FOR REPEATED TRAPS AND OVERFLOW OF STACK

```

2549 ;RD GIVES LOCATION OF STACK ERROR
2550 011320 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
; "JMP 2R0" (000110) TO GET A
; SCOPE LOOP AND HIT CONTINUE
2551
2552 011322 000402 07: BR 06
2553 011324 000176 000000 05: JMP 2(6) ;RETURN FROM STACK
2554 011330 010700 06: SCOPE ;OVERFLOW TRAP
2555
;*****
; *TEST 107 TEST TRACE TRAP WHEN STACK IS IN YELLOW AREA
;*****
2556
2557 011332 005237 000304 TST107: INC 2#TESTN
2558 011336 005037 021414 CLR TEMP
2559 011342 012706 000400 MOV #400,X6 ;SET UP STACK POINTER
2560 011346 012737 011532 000014 MOV #TRTC,TRTVEC ;LOAD TRT VECTOR
2561 011354 005037 000016 CLR TRTVEC+2
2562 011360 012737 011416 000004 MOV #TRTB,ERRVEC ;LOAD OVFLW. ERROR VECTOR
2563 011366 005037 000006 CLR ERRVEC+2
2564 011372 013737 011410 021416 MOV TRTA,TEMP+2 ;SAVE FIRST WORD OF INST. AT TRTA
2565 011400 012737 000003 011410 MOV #BPT,TRTA ;REPLACE WITH TRT INSTRUCTION
2566 011406 000277 SCC ;SET CONDITION CODES
2567 011410 005137 021414 TRTA: COM TEMP ;NOTE: THIS INSTRUCTION WAS REPLACED BY TRT
2568 011414 000535 TRTAA: BR TRTF
2569 011416 022737 000017 000376 TRTB: CMP #17,376 ;WERE CONDITION CODES SAVED
2570 011424 001405 BEQ 1$
2571
2572 011426 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2573 011430 012713 000204 MOV #204,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2574 011434 000000 HLT ;ERROR! CONDITION CODES NOT SAVED
2575 011436 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
; "JMP 2R0" (000110) TO GET A
; SCOPE LOOP AND HIT CONTINUE
; WAS RETURN FROM TRT SAVED
2576
2577 011440 022737 011412 000374 1$: CMP #TRTA+2,374
2578 011446 001406 BEQ 2$
2579
2580 011450 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2581 011452 012713 000205 MOV #205,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2582 011456 000000 HLT ;ERROR! RETURN PC NOT SAVED
2583 011460 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
; "JMP 2R0" (000110) TO GET A
; SCOPE LOOP AND HIT CONTINUE
; EXIT TEST
; WAS TRT STATUS SAVED?
2584
2585 011462 000523 BR TRTEX
2586 011464 005737 000372 2$: TST 372
2587 011470 001405 BEQ 3$
2588
2589 011472 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2590 011474 012713 000206 MOV #206,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2591 011500 000000 HLT ;ERROR! TRT STATUS (TRTVEC+2) NOT SAVED
2592 011502 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
; "JMP 2R0" (000110) TO GET A
; SCOPE LOOP AND HIT CONTINUE
; TRT VECTOR SAVED
2593
2594 011504 022737 011532 000370 3$: CMP #TRTC,370
2595 011512 001406 BEQ 4$
2596
2597 011514 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2598 011516 012713 000207 MOV #207,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
  
```

2605	011522	000000				HLT			: ERROR! TRT VECTOR (TRTVEC) NOT SAVED
2606	011524	000240				LOOP			: REPLACE THIS INSTRUCTION BY A
2607									: "JMP JRO" (000110) TO GET A
2608									: SCOPE LOOP AND HIT CONTINUE
2609	011526	000501				BR	TRTEX		: EXIT TEST
2610	011530	000002			4\$:	RTI			: RETURN FROM OVERFLOW TRAP
2611									
2612	011532	010605			TRTC:	MOV	%6,%5		: GET STACK POINTER
2613	011534	162715	000002			SUB	#2,(5)		: POINT PC TO TRAPPED INSTRUCTION
2614	011540	013735	021416			MOV	TEMP+2,(5)+		: RESTORE ORIGINAL INSTRUCTION (COM TEMP)
2615	011544	052715	000020			BIS	#20,(5)		: SET 'T' BIT ON STACK
2616	011550	012737	011566	000004		MOV	#TRTD,ERRVEC		: CHANGE ERROR VECTOR
2617	011556	012737	011702	000014		MOV	#TRTE,TBITVEC		: LOAD 'T' BIT VECTOR
2618	011564	000006				RTT			: GO DO ORIGINAL INSTRUCTION AND TRAP
2619									: ('T' BIT TRAP) WHEN FINISHED
2620									
2621									: CHECK THE STACK
2622	011566	022737	000031	000376	TRTD:	CMP	#31,376		: IS 'N', 'C', 'T' BITS SET ON THE STACK
2623	011574	001405				BEQ	1\$: (RESULT OF 'COM TEMP' IN TRTA
2624									
2625	011576	005212				INC	(R2)		: SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2626	011600	012713	000210			MOV	#210,(R3)		: MOVE TO MAIL BOX (\$FATAL) THE ERROR
2627	011604	000000				HLT			: ERROR! STATUS NOT SAVED
2628	011606	000240				LOOP			: REPLACE THIS INSTRUCTION BY A
2629									: "JMP JRO" (000110) TO GET A
2630									: SCOPE LOOP AND HIT CONTINUE
2631	011610	022737	011414	000374	1\$:	CMP	#TRTAA,374		: WAS RETURN PC FROM 'T' TRAP SAVED?
2632	011616	001406				BEQ	2\$		
2633									
2634	011620	005212				INC	(R2)		: SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2635	011622	012713	000211			MOV	#211,(R3)		: MOVE TO MAIL BOX (\$FATAL) THE ERROR
2636	011626	000000				HLT			: ERROR! RETURN PC NOT SAVED
2637	011630	000240				LOOP			: REPLACE THIS INSTRUCTION BY A
2638									: "JMP JRO" (000110) TO GET A
2639									: SCOPE LOOP AND HIT CONTINUE
2640	011632	000437				BR	TRTEX		: EXIT TEST
2641	011634	005737	000372		2\$:	TST	372		: 'T' BIT TRAP STATUS SAVED?
2642	011640	001405				BEQ	3\$		
2643									
2644	011642	005212				INC	(R2)		: SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2645	011644	012713	000212			MOV	#212,(R3)		: MOVE TO MAIL BOX (\$FATAL) THE ERROR
2646	011650	000000				HLT			: ERROR! 'T' BIT STATUS (TBITVEC+2) NOT SAVED
2647	011652	000240				LOOP			: REPLACE THIS INSTRUCTION BY A
2648									: "JMP JRO" (000110) TO GET A
2649									: SCOPE LOOP AND HIT CONTINUE
2650	011654	022737	011702	000370	3\$:	CMP	#TRTE,370		: 'T' BIT VECTOR SAVED (TBITVRC)
2651	011662	001406				BEQ	4\$		
2652									
2653	011664	005212				INC	(R2)		: SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2654	011666	012713	000213			MOV	#213,(R3)		: MOVE TO MAIL BOX (\$FATAL) THE ERROR
2655	011672	000000				HLT			: ERROR! 'T' BIT VECTOR NOT SAVED
2656	011674	000240				LOOP			: REPLACE THIS INSTRUCTION BY A
2657									: "JMP JRO" (000110) TO GET A
2658									: SCOPE LOOP AND HIT CONTINUE
2659	011676	000415				BR	TRTEX		: EXIT TEST
2660	011700	000002			4\$:	RTI			: RETURN FROM ERROR TRAP

```

2661
2662 011702 005037 000375 TRTE: CLR 376 ;CLEAR SAVED 'T' BIT
2663 011706 000002 RTI ;RETURN FROM 'T' BIT TRAP
2664
2665 011710 022737 177777 (21414 TRTF: CMP #-1,TEMP ;WAS ORIGINAL INSTRUCTION EXECUTED?
2666 011716 001405 BEQ TRTEX
2667
2668 011720 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2669 011722 012713 000214 IS: MOV #214,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2670 011726 000000 HLT ;TRACED INSTRUCTION (COM TEMP) NOT EXEC.
2671 011730 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
2672 ;"JMP JRO" (000110) TO GET A
2673 ;SCOPE LOOP AND HIT CONTINUE
2674 011732 012737 000016 (000014 TRTEX: MOV #TRTVEC+2,TRTVEC;RESTORE TRAP
2675 011740 012737 000006 000004 MOV #6,4 ;VECTORS
2676 011746 010700 SCOPE
2677
2678 ;*****
2679 ;*TEST 110 TEST YELLOW ZONE STACK VIOLATION, USING INDEX MODE (6) & R6. (R6.GT.4
2680 ;*****
2681 011750 005237 000304 TST110: INC #STESTN
2682 011754 012706 001000 INDEX: MOV #1000,%6 ;SET UP STACK POINTER
2683 011760 012737 012010 000004 MOV #IND1,4
2684 011766 005037 000006 CLR 6
2685 011772 005066 177340 CLR -440(6) ;FINAL ADDRESS IS 340
2686
2687 011776 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2688 012000 012713 000215 MOV #215,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2689 012004 000000 HLT ;ERROR! STACK OVERFLOW TRAP DID NOT OCCUR
2690 012006 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
2691 ;"JMP JRO" (000110) TO GET A
2692 ;SCOPE LOOP AND HIT CONTINUE
2693 012010 010700 IND1: SCOPE
2694
2695 ;TEST THAT THE TRAP SEQUENCE IS EXECUTED PROPERLY ON OVERFLOW.
2696 012012 012706 001000 MOV #1000,%6
2697 012016 012737 012064 000004 MOV #IND2,4
2698 012024 005037 000776 CLR 776
2699 012030 005037 000774 CLR 774
2700 012034 012737 000001 000376 MOV #1,376 ;PRE SET DEST ADDRESS
2701 012042 012766 000000 177376 MOV #0,-402(6) ;FINAL ADDRESS IS 376
2702 012050 IND2B:
2703
2704 012050 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2705 012052 012713 000216 IND2A: MOV #216,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2706 012056 000000 HLT ;ERROR! FAILED TO TRAP
2707 012060 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
2708 ;"JMP JRO" (000110) TO GET A
2709 ;SCOPE LOOP AND HIT CONTINUE
2710 012062 000442 BR IND2X
2711 012064 023727 000776 000004 IND2: CMP 776,#4
2712 012072 001405 BEQ 15 ;STATUS SAVED ON THE STACK (Z BIT SET)
2713 ;(RESULT OF MOV #0,-402(6))
2714 012074 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2715 012076 012713 000217 MOV #217,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2716 012102 000000 HLT ;ERROR! INCORRECT STATUS SAVED
  
```


F05

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 57
 DOKDBA.P11 17-FEB-77 08:26

T112 TEST RED ZONE STACK VIOLATION & ABORT OF THE VIOLATING INSTRUCTION

2829										; "JMP JRO" (000110) TO GET A
2830										; SCOPE LOOP AND HIT CONTINUE
2831	012454	022737	177777	000336	1\$:	CMP	#-1,336			; WAS CLEAR INSTRUCTION ABORTED?
2832	012462	001405				BEQ	2\$			
2833										
2834	012464	005212				INC	(R2)			; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2835	012466	012713	000232				MOV	#232, (R3)		; MOVE TO MAIL BOX (\$FATAL) THE ERROR
2836	012472	000000				HLT				
2837	012474	000240				LOOP				
2838										; REPLACE THIS INSTRUCTION BY A
2839										; "JMP JRO" (000110) TO GET A
2840	012476	022737	000004	000002	2\$:	CMP	#4,2			; SCOPE LOOP AND HIT CONTINUE
2841	012504	001405				BEQ	3\$; WAS PROPER STATUS SAVED? (Z SET)
2842										; RESULT OF CLR -442(6)
2843	012506	005212				INC	(R2)			; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2844	012510	012713	000233				MOV	#233, (R3)		; MOVE TO MAIL BOX (\$FATAL) THE ERROR
2845	012514	000000				HLT				
2846										; ERROR! INCORRECT STATUS WAS SAVED
2847										; THOUGH THE INSTR. (CLR) WAS ABORTED
2848										; THE Z BIT WAS SET BECAUSE OF CLEAR
2849	012516	000240				LOOP				
2850										; REPLACE THIS INSTRUCTION BY A
2851										; "JMP JRO" (000110) TO GET A
2852	012520	022737	012426	000000	3\$:	CMP	#IND48,0			; SCOPE LOOP AND HIT CONTINUE
2853	012526	001405				BEQ	4\$			
2854										; WAS RETURN PC SAVED?
2855	012530	005212				INC	(R2)			; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2856	012532	012713	000234				MOV	#234, (R3)		; MOVE TO MAIL BOX (\$FATAL) THE ERROR
2857	012536	000000				HLT				
2858	012540	000240				LOOP				
2859										; REPLACE THIS INSTRUCTION BY A
2860										; "JMP JRO" (000110) TO GET A
2861	012542	005737	000776				TST	776		; SCOPE LOOP AND HIT CONTINUE
2862	012546	001405				BEQ	5\$; WAS NOTHING PUT ON THE OLD STACK?
2863										
2864	012550	005212				INC	(R2)			; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2865	012552	012713	000235				MOV	#235, (R3)		; MOVE TO MAIL BOX (\$FATAL) THE ERROR
2866	012556	000000				HLT				
2867	012560	000240				LOOP				
2868										; ERROR! RED TRAP DID NOT STACK AT 2 & 0
2869										; REPLACE THIS INSTRUCTION BY A
2870	012562	010700				SCOPE				
2871										; "JMP JRO" (000110) TO GET A
2872										; SCOPE LOOP AND HIT CONTINUE
2873										; *****
2874										; *TEST 113 TEST THAT INDEX MODE USING R6 IN SOURCE DOES NOT TRAP
2875	012564	005237	000304				TST113: INC	#\$TESTN		; *****
2876	012570	012706	001000				MOV	#1000,%6		; SET UP STACK
2877	012574	012737	012626	000004				MOV	#IND5,4	; SET UP ERROR TRAP
2878	012602	012737	177777	000350				MOV	#-1,350	
2879	012610	016637	177350	021414				MOV	-430(6),TEMP	
2880	012616	000405				BR	IN08			
2881										
2882	012620	005212				INC	(R2)			; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
2883	012622	012713	000236				MOV	#236, (R3)		; MOVE TO MAIL BOX (\$FATAL) THE ERROR
2884	012626	000000				HLT				

H05

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 59
 DOKDBA.P11 17-FEB-77 08:26

T116 THIS TEST IS THE SAME AS ABOVE EXCEPT THE ADDRESS IS 'RED'

```

2941 013012 010700          IND12: SCOPE
2942
2943          ;*****
2944          ;*TEST 117 TEST RED ZONE VIOLATION USING EMT INSTRUCTION
2945          ;*****
2946 013014 005237 000304          TST117: INC      @#STESTN
2947 013020 012706 000340          MOV      #340,%6          ;SET UP STACK POINTER IN 'RED' AREA
2948 013024 012737 013112 000030          MOV      @RED1,EMTVEC
2949 013032 005037 000032          CLR      EMTVEC+2
2950 013036 012737 013116 000004          MOV      @RED1A,4
2951 013044 005037 000006          CLR      6
2952 013050 005037 000000          CLR      0
2953 013054 012737 136336 000336          MOV      #136336,336          ;PRESET 'RED' LOCATION
2954 013062 012737 177777 000002          MOV      #-1,2          ;N BIT SET
2955 013070 104000
2956 013072          RED1C:
2957
2958 013072 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2959 013074 012713 000242          MOV      #242,(R3)          ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2960 013100 000000          RED1B: HLT
2961 013102 000240          LOOP
2962          ;REPLACE THIS INSTRUCTION BY A
2963          ;"JMP @RO" (000110) TO GET A
2964          ;SCOPE LOOP AND HIT CONTINUE
2965 013104 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2966 013106 012713 000243          MOV      #243,(R3)          ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2967 013112 000000          RED1: HLT
2968 013114 000240          LOOP
2969          ;REPLACE THIS INSTRUCTION BY A
2970          ;"JMP @RO" (000110) TO GET A
2971          ;SCOPE LOOP AND HIT CONTINUE
2972          ;WAS (NEW) STATUS SAVED?
2973
2974 013116 022737 000000 000002          RED1A: CMP      #0,2
2975 013124 001405          BEQ      1$
2976
2977 013126 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2978 013130 012713 000244          MOV      #244,(R3)          ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2979 013134 000000          HLT
2980          ;CORRECT (NEW) PSW WAS NOT SAVED
2981          ;ON A RED ZONE TRAP (CAUSED BY EMT)
2982          ;UPON EXECUTING EMT RED ZONE OCCURED
2983          ;THE 11/60 SAVES THE "NEW" PSW WHICH
2984          ;WOULD HAVE BEEN LOADED BY THE EMT.
2985          ;11/40 SAVES THE PSW THAT WAS CURRENT AT THE
2986          ;TIME EMT WAS EXECUTED.)
2987 013136 000240          LOOP
2988          ;REPLACE THIS INSTRUCTION BY A
2989          ;"JMP @RO" (000110) TO GET A
2990          ;SCOPE LOOP AND HIT CONTINUE
2991          ;WAS RETURN PC SAVED
2992          ;(EMTVEC)
2993
2994 013140 022737 013072 000000          1$: CMP      #RED1C,0
2995 013146 001405          BEQ      2$
2996
2997 013150 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2998 013152 012713 000245          MOV      #245,(R3)          ;MOVE TO MAIL BOX ($FATAL) THE ERROR
2999 013156 000000          HLT
3000 013160 000240          LOOP
3001          ;REPLACE THIS INSTRUCTION BY A
3002          ;"JMP @RO" (000110) TO GET A
3003          ;SCOPE LOOP AND HIT CONTINUE
3004          ;STACK POINTER=0?
3005 013162 005706          2$: TST      %6
3006 013164 001405          BEQ      3$
  
```

```

2997
2998 013166 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
2999 013170 012713 000246   MOV      #246,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3000 013174 000000          HLT
3001 013176 000240          LOOP
3002
3003
3004 013200 022737 136336 000336 3$:  CMP      #136336,336 ;REPLACE THIS INSTRUCTION BY A
3005 013206 001405          BEQ      4$          ;"JMP 2R0" (000110) TO GET A
3006
3007 013210 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3008 013212 012713 000247   MOV      #247,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3009 013216 000000          HLT
3010 013220 000240          LOOP
3011
3012
3013 013222 012737 000032 000030 4$:  MOV      #EMTVEC+2,EMTVEC ;REPLACE THIS INSTRUCTION BY A
3014 013230 010700          SCOPE
3015
3016
3017
3018
3019 013232 005237 000304          TST120: INC      #TESTN
3020 013236 012706 000200          MOV      #200,X6
3021 013242 012737 176176 000176   MOV      #176176,176 ;SET UP STACK IN 'RED' AREA
3022 013250 012737 013326 000034   MOV      #RED2,TRAPVEC ;PRE SET 'RED' LOCATION
3023 013256 005037 000036   CLR      TRAPVEC+2
3024 013262 005037 000006   CLR      6
3025 013266 005037 000000   CLR      0
3026 013272 012737 013332 000004   MOV      #RED2A,4
3027 013300 005037 000002   CLR      2          ;Z BIT SET
3028 013304 104400          TRAP
3029 013306          RED2C:
3030
3031 013306 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3032 013310 012713 000250   MOV      #250,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3033 013314 000000          HLT
3034 013316 000240          LOOP
3035
3036
3037
3038 013320 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3039 013322 012713 000251   MOV      #251,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3040 013326 000000          HLT
3041 013330 000240          LOOP
3042
3043
3044 013332 022737 000000 000002 RED2A: CMP      #0,2
3045 013340 001405          BEQ      1$
3046
3047 013342 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3048 013344 012713 000252   MOV      #252,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3049 013350 000000          HLT
3050
3051
3052
  
```


K05

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 62
 DOKDBA.P11 17-FEB-77 08:26 T121 TEST RED ZONE VIOLATION USING "IOT" INSTRUCTION

3109	013526	005212				INC	(R2)	; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
3110	013530	012713	000257			MOV	#257, (R3)	; MOVE TO MAIL BOX (\$FATAL) THE ERROR
3111	013534	000000			RED3:	HLT		; DID NOT ABORT IOT
3112	013536	000240				LOOP		; REPLACE THIS INSTRUCTION BY A
3113								; "JMP 2R0" (000110) TO GET A
3114								; SCOPE LOOP AND HIT CONTINUE
3115	013540	022737	000000	000002	RED3A:	CMP	#0, 2	; WAS STATUS SAVED?
3116	013546	001405				BEQ	15	
3117								
3118	013550	005212				INC	(R2)	; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
3119	013552	012713	000260			MOV	#260, (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". 11/60 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	000240				LOOP		; REPLACE THIS INSTRUCTION BY A
3127								; "JMP 2R0" (000110) TO GET A
3128								; SCOPE LOOP AND HIT CONTINUE
3129	013562	022737	013514	000000	15:	CMP	#RED3C, 0	; WAS RETURN PC SAVED?
3130	013570	001405				BEQ	25	; (IOTVEC)
3131								
3132	013572	005212				INC	(R2)	; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
3133	013574	012713	000261			MOV	#261, (R3)	; MOVE TO MAIL BOX (\$FATAL) THE ERROR
3134	013600	000000				HLT		
3135	013602	000240				LOOP		; REPLACE THIS INSTRUCTION BY A
3136								; "JMP 2R0" (000110) TO GET A
3137								; SCOPE LOOP AND HIT CONTINUE
3138	013604	005706			25:	TST	%6	
3139	013606	001405				BEQ	35	
3140								
3141	013610	005212				INC	(R2)	; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
3142	013612	012713	000262			MOV	#262, (R3)	; MOVE TO MAIL BOX (\$FATAL) THE ERROR
3143	013616	000000				HLT		
3144	013620	000240				LOOP		; REPLACE THIS INSTRUCTION BY A
3145								; "JMP 2R0" (000110) TO GET A
3146								; SCOPE LOOP AND HIT CONTINUE
3147	013622	012737	000022	000020	35:	MOV	#IOTVEC+2, IOTVEC	
3148	013630	010700				SCOPE		
3149								
3150								
3151								; *****
3152								; *TEST 122 TEST RED ZONE VIOLATION UPON TRACE TRAP
3153	013632	005237	000304		TST122:	INC	#TESTN	
3154	013636	005006				CLR	%6	
3155	013640	012737	013716	000014		MOV	#RED4, TRTVEC	
3156	013646	005037	000016			CLR	TRTVEC+2	
3157	013652	005037	000006			CLR	6	
3158	013656	012737	013722	000004		MOV	#RED4A, 4	
3159	013664	005037	000000			CLR	0	
3160	013670	005037	000002			CLR	2	
3161	013674	000003				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 ($FATAL) THE ERROR
3166 013704 000000          RED4B:  HLT              ;DID NOT OVERFLOW
3167 013706 000240          LOOP              ;REPLACE THIS INSTRUCTION BY A
3168                                     ;"JMP 2R0" (000110) TO GET A
3169                                     ;SCOPE LOOP AND HIT CONTINUE
3170
3171 013710 005212          INC      (R2)        ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3172 013712 012713 000264          RED4:  MOV      #264,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3173 013716 000000          HLT              ;DID NOT ABORT TRT
3174 013720 000240          LOOP              ;REPLACE THIS INSTRUCTION BY A
3175                                     ;"JMP 2R0" (000110) TO GET A
3176                                     ;SCOPE LOOP AND HIT CONTINUE
3177 013722 022737 000000 000002 RED4A:  CMP      #0,2        ;WAS CORRECT ? SW SAVED?
3178 013730 001405          BEQ
3179
3180 013732 005212          INC      (R2)        ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3181 013734 012713 000265          MOV      #265,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3182 013740 000000          HLT              ;CORRECT STATUS WORD WAS NOT SAVED
3183                                     ;UPON A RED ZONE TRAP CAUSED BY A BPT
3184                                     ;NEW PSH TAP WOULD BE LOADED BY BPT
3185                                     ;SHOULD BE SAVED.
3186 013742 000240          LOOP              ;REPLACE THIS INSTRUCTION BY A
3187                                     ;"JMP 2R0" (000110) TO GET A
3188                                     ;SCOPE LOOP AND HIT CONTINUE
3189 013744 022737 013676 000000 15:  CMP      #RED4C,0    ;WAS RETURN PC SAVED?
3190 013752 001405          BEQ      25
3191
3192 013754 005212          INC      (R2)        ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3193 013756 012713 000266          MOV      #266,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3194 013762 000000          HLT              ;REPLACE THIS INSTRUCTION BY A
3195 013764 000240          LOOP              ;"JMP 2R0" (000110) TO GET A
3196                                     ;SCOPE LOOP AND HIT CONTINUE
3197
3198 013766 005706          25:  TST      %6
3199 013770 001405          BEQ      35
3200
3201 013772 005212          INC      (R2)        ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3202 013774 012713 000267          MOV      #267,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3203 014000 000000          HLT              ;REPLACE THIS INSTRUCTION BY A
3204 014002 000240          LOOP              ;"JMP 2R0" (000110) TO GET A
3205                                     ;SCOPE LOOP AND HIT CONTINUE
3206
3207 014004 012737 000016 000014 35:  MOV      #TRTVEC+2,TRTVEC
3208 014012 010700          SCOPE
3209
3210                                     ;*****
3211                                     ;*TEST 123 TEST TRANSITION FROM 'YELLOW' TO 'RED' AREAS
3212                                     ;*****
3213 014014 005237 000304          TST123: INC      #TESTN
3214                                     ;THE TRANSITION OCCURS AFTER THE EMT HAS PUSHED ITS RETURN ADDRESS AND STATUS.
3215 014020 012706 000344          MOV      #344,%6    ;SET UP STACK TO ALLOW 2 'PUSHES'
3216 014024 012737 014124 000030          MOV      #REDS,EMTVEC ;LOAD EMT VECTOR
3217 014032 005037 000032          CLR      EMTVEC+2   ;AND STATUS
3218 014036 012737 014132 000004          MOV      #REDSA,4   ;LOAD OVERFLOW VECTOR
3219 014044 012737 000001 000006          MOV      #1,6
3220 014052 012737 000017 000002          MOV      #17,2

```


M05

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 64
 DLKDBA.P11 17-FEB-77 08:26 T123 TEST TRANSITION FROM 'YELLOW' TO 'RED' AREAS

3221	014060	005037	000000			CLR	0	
3222	014064	012737	136336	000336		MOV	#136336,336	;PRE SET 'RED' LOCATION
3223	014072	012737	000004	177776		MOV	#4,CC	
3224	014100	104000				EMT		
3225	014102				REDESC:			
3226								
3227	014102	005212				INC	(R2)	;SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
3228	014104	012713	000270		REDSB:	MOV	#270,(R3)	;MOVE TO MAIL BOX (\$FATAL) THE ERROR
3229	014110	000000				HLT		
3230	014112	000240				LOOP		;REPLACE THIS INSTRUCTION BY A
3231								; "JMP 2RD" (000110) TO GET A
3232								; SCOPE LOOP AND HIT CONTINUE
3233	014114	000240				NOP		
3234								
3235	014116	005212				INC	(R2)	;SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
3236	014120	012713	000271		REDS:	MOV	#271,(R3)	;MOVE TO MAIL BOX (\$FATAL) THE ERROR
3237	014124	000000				HLT		
3238	014126	000240				LOOP		;REPLACE THIS INSTRUCTION BY A
3239								; "JMP 2RD" (000110) TO GET A
3240								; SCOPE LOOP AND HIT CONTINUE
3241	014130	000240				NOP		
3242	014132	022737	000004	000342	REDSA:	CMP	#4,342	; WAS STATUS SAVED?
3243	014140	001405				BEQ	15	
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 2RD" (000110) TO GET A
3250								; SCOPE LOOP AND HIT CONTINUE
3251	014154	022737	014102	000340	15:	CMP	#REDESC,340	; WAS RETURN PC SAVED?
3252	014162	001405				BEQ	25	
3253								
3254	014164	005212				INC	(R2)	;SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
3255	014166	012713	000273			MOV	#273,(R3)	;MOVE TO MAIL BOX (\$FATAL) THE ERROR
3256	014172	000000				HLT		; ERROR! RETURN PC NOT SAVED
3257	014174	000240				LOOP		;REPLACE THIS INSTRUCTION BY A
3258								; "JMP 2RD" (000110) TO GET A
3259								; SCOPE LOOP AND HIT CONTINUE
3260	014176	022737	136336	000336	25:	CMP	#136336,336	; WAS 'RED' LOCATION LEFT UNDISTURBED?
3261	014204	001405				BEQ	35	
3262								
3263	014206	005212				INC	(R2)	;SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
3264	014210	012713	000274			MOV	#274,(R3)	;MOVE TO MAIL BOX (\$FATAL) THE ERROR
3265	014214	000000				HLT		; ERROR! 'RED' AREA WAS CHANGED
3266	014216	000240				LOOP		;REPLACE THIS INSTRUCTION BY A
3267								; "JMP 2RD" (000110) TO GET A
3268								; SCOPE LOOP AND HIT CONTINUE
3269	014220	022737	000001	000002	35:	CMP	#1,2	; WAS CORRECT PSW SAVED?
3270	014226	001405				BEQ	45	
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 PSW WAS NOT SAVED WHEN
3275								; A RED ZONE OCCURED WHILE TRYING
3276								; TO SERVICE YELLOW ZONE. AFTER THE EMT


```

3389 014550 000000          HLT          ;OVERFLOW TRAP DID NOT SAVE RETURN PC
3390 014552 000240          LOOP        ;REPLACE THIS INSTRUCTION BY A
3391                                     ;"JMP 2RO" (000110) TO GET A
3392                                     ;SCOPE LOOP AND HIT CONTINUE
3393 014554 022737 136336 000336 45:  CMP      #136336,336 ;WAS 'RED' LOCATION UNDISTURBED?
3394 014562 001405          BEQ      55
3395
3396 014564 005212          INC      (R2)      ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3397 014566 012713 000307  MOV      #307,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3398 014572 000000          HLT          ;ERROR! 'RED' LOCATION WAS CHANGED
3399 014574 000240          LOOP        ;REPLACE THIS INSTRUCTION BY A
3400                                     ;"JMP 2RO" (000110) TO GET A
3401                                     ;SCOPE LOOP AND HIT CONTINUE
3402 014576 010700          55:  SCOPE
3403
3404                                     ;*****
3405                                     ;*TEST 125 TEST TRANSITION FROM 'YELLOW' TO 'RED' ZONE
3406                                     ;*****
3407 014600 005237 000304  TST125: INC      #0$TESTN
3408                                     ;THIS TEST IS THE SAME AS ABOVE EXCEPT THAT THE TRANSITION IS AFTER THE
3409                                     ;OVERFLOW PUSHES ONE WORD.
3410 014604 012706 000344  MOV      #344,%6
3411 014610 012737 014676 000004  MOV      #RED7A,4 ;LOAD OVERFLOW VECTOR
3412 014616 005037 000006  CLR      6
3413 014622 005037 000342  CLR      342
3414 014626 012737 136336 000336  MOV      #136336,336 ;PRE SET 'RED' LOCATION
3415 014634 012705 000007  MOV      #7,%5 ;PRE SET RS
3416 014640 012737 000357 177776  MOV      #357,CC ;PRE SET STATUS
3417 014646 004537 014672  JSR      5,RED7 ;CAUSE OVERFLOW
3418 014652          RED7C:
3419
3420 014652 005212          INC      (R2)      ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3421 014654 012713 000310  MOV      #310,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3422 014660 000000          HLT          ;ERROR DID NOT OVERFLOW TRAP
3423 014662 000240          LOOP        ;REPLACE THIS INSTRUCTION BY A
3424                                     ;"JMP 2RO" (000110) TO GET A
3425                                     ;SCOPE LOOP AND HIT CONTINUE
3426
3427 014664 005212          INC      (R2)      ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3428 014666 012713 000311  MOV      #311,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3429 014672 000000          HLT          ;ERROR! DID NOT TRAP BEFORE JSR
3430 014674 000240          LOOP        ;REPLACE THIS INSTRUCTION BY A
3431                                     ;"JMP 2RO" (000110) TO GET A
3432                                     ;SCOPE LOOP AND HIT CONTINUE
3433 014676 022737 000007 000342  RED7A: CMP      #7,342
3434 014704 001405          BEQ      15
3435
3436 014706 005212          INC      (R2)      ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3437 014710 012713 000312  MOV      #312,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3438 014714 000000          HLT          ;ERROR! RS NOT SAVED ON THE STACK
3439 014716 000240          LOOP        ;REPLACE THIS INSTRUCTION BY A
3440                                     ;"JMP 2RO" (000110) TO GET A
3441                                     ;SCOPE LOOP AND HIT CONTINUE
3442 014720 022705 014652          15:  CMP      #RED7C,%5 ;DOES RS CONTAIN RETURN PC?
3443 014724 001405          BEQ      25
3444

```

```

3445 014726 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3445 014730 012713 000313  MOV      #313,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3445 014734 000000          HLT                    ;ERROR! JSR DID NOT LOAD RS
3445 014736 000240          LOOP                   ;REPLACE THIS INSTRUCTION BY A
3445                                ;"JMP 2R0" (000110) TO GET A
3445                                ;SCOPE LOOP AND HIT CONTINUE
3451 014740 022737 000357 000340 2$:  CMP      #357,340    ;WAS STATUS SAVED?
3451 014746 001405          BEQ      3$
3454 014750 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3454 014752 012713 000314  MOV      #314,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3454 014756 000000          HLT                    ;ERROR! STATUS NOT SAVED
3454 014760 000240          LOOP                   ;REPLACE THIS INSTRUCTION BY A
3454                                ;"JMP 2R0" (000110) TO GET A
3454                                ;SCOPE LOOP AND HIT CONTINUE
3460 014762 022737 000000 000002 3$:  CMP      #0,2
3460 014770 001405          BEQ      4$
3463 014772 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3463 014774 012713 000315  MOV      #315,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3463 015000 000000          HLT                    ;CORRECT PSW WAS NOT SAVED WHEN
3463                                ;A RED ZONE OCCURED WHILE TRYING
3463                                ;TO SERVICE YELLOW ZONE. AFTER THE JSR
3463                                ;IS EXECUTED THE PENDING YELLOW ZONE OCCURS
3463                                ;(AND YELLOW ZONE PSW IS LOADED), BUT TRYING
3463                                ;TO PUSH THE STACK CAUSES RED ZONE AND
3463                                ;THE NEW PSW (YELLOW ZONE'S) SHOULD
3463                                ;BE SAVED ON THE STACK
3463                                ;REPLACE THIS INSTRUCTION BY A
3463                                ;"JMP 2R0" (000110) TO GET A
3463                                ;SCOPE LOOP AND HIT CONTINUE
3473 015002 000240          LOOP                   ;RETURN PC SAVED?
3476 015004 022737 014672 000000 4$:  CMP      #RED7,0
3476 015012 001405          BEQ      5$
3479 015014 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3479 015016 012713 000316  MOV      #316,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3479 015022 000000          HLT                    ;ERROR! RETURN PC NOT SAVED
3479 015024 000240          LOOP                   ;REPLACE THIS INSTRUCTION BY A
3479                                ;"JMP 2R0" (000110) TO GET A
3479                                ;SCOPE LOOP AND HIT CONTINUE
3485 015026 022737 136336 000336 5$:  CMP      #136336,336
3485 015034 001405          BEQ      6$
3488 015036 005212          INC      (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3488 015040 012713 000317  MOV      #317,(R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3488 015044 000000          HLT                    ;ERROR! 'RED' LOCATION WAS CHANGED
3488 015046 000240          LOOP                   ;REPLACE THIS INSTRUCTION BY A
3488                                ;"JMP 2R0" (000110) TO GET A
3488                                ;SCOPE LOOP AND HIT CONTINUE
3494 015050 010700          6$:  SCOPE
3496                                ;*****
3496                                ;*TEST 126 TEST THAT A TTY INTERRUPT CAUSES AN OVERFLOW TRAP
3496                                ;*****
3499 015052 005237 000304  TST126: INC      #TESTN
3500 015056 012737 000340 177776  MOV      #340,STATUS ;LOCK OUT INTERRUPT

```

E06

MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 69
 DOKDBA.P11 17-FEB-77 08:26

T126 TEST THAT A TTY INTERRUPT CAUSES AN OVERFLOW TRAP

3501	015064	012706	000400		MOV	#400,%6	;SET UP STACK TO OVERFLOW
3502	015070	012737	015142	000004	MOV	#TDEC7,4	;SET UP OVERFLOW TRAP
3503	015076	012737	015136	000064	MOV	#TDEC8,64	;SET UP INTERRUPT VECTOR
3504	015104	012737	000100	177564	MOV	#100,TTCSR	;SET INTERRUPT ENABLE
3505	015112	005037	177776		CLR	STATUS	;ALLOW INTERRUPT TO OCCUR
3506							
3507	015116	005212			INC	(R2)	;SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
3508	015120	012713	000320		MOV	#320,(R3)	;MOVE TO MAIL BOX (\$FATAL) THE ERROR
3509	015124	000000			HLT		;NO INTERRUPT OCCURRED
3510	015126	000240			LOOP		;REPLACE THIS INSTRUCTION BY A
3511							; "JMP 2R0" (000110) TO GET A
3512							; SCOPE LOOP AND HIT CONTINUE
3513							
3514	015130	005212			INC	(R2)	;SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
3515	015132	012713	000321		MOV	#321,(R3)	;MOVE TO MAIL BOX (\$FATAL) THE ERROR
3516	015136	000000			TDEC8: HLT		;TRAP FLAG OVERFLOW DID NOT OCCUR
3517	015140	000240			LOOP		;REPLACE THIS INSTRUCTION BY A
3518							; "JMP 2R0" (000110) TO GET A
3519							; SCOPE LOOP AND HIT CONTINUE
3520	015142	005037	177564		TDEC7: CLR	TTCSR	;CLEAR INTERRUPT ENABLE
3521	015146	010700			SCOPE		

```

3522
3523
3524
3525
3526 015150 005237 000304 TST127: INC @#STESTN
3527 015154 012706 021774 MOV #BUFF,%6 ;SET UP STACK POINTER
3528 015160 012737 015214 000004 MOV #R7TR1,4 ;RETURN FROM TRAP
3529 015166 012707 000001 MOV #1,%7 ;PC EQUALS ONE
3530
3531 015172 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3532 015174 012713 000322 MOV #322,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3533 015200 000000 HLT
3534 015202 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
3535 ;"JMP @RO" (000110) TO GET A
3536 ;SCOPE LOOP AND HIT CONTINUE
3537
3538 015204 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3539 015206 012713 000323 MOV #323,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3540 015212 000000 HLT
3541 015214 022737 000003 021770 R7TR1: CMP #3,BUFF-4
3542 015222 001405 BEQ 15 ;CORRECT PC WAS NOT SAVED ON STACK
3543
3544 015224 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3545 015226 012713 000324 MOV #324,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3546 015232 000000 HLT
3547 015234 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
3548 ;"JMP @RO" (000110) TO GET A
3549 ;SCOPE LOOP AND HIT CONTINUE
3550
3551 015236 010700 15: SCOPE
3552
3553
3554
3555
3556 015240 005237 000304 TST130: INC @#STESTN
3557 015244 012706 021774 MOV #BUFF,%6 ;STACK POINTER
3558 015250 012737 015302 000004 MOV #R7TR2,4
3559 015256 005207 INC %7 ;PC BECOMES ODD
3560 015260 R7TR2A:
3561
3562 015260 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3563 015262 012713 000325 MOV #325,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3564 015266 000000 HLT
3565 015270 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
3566 ;"JMP @RO" (000110) TO GET A
3567 ;SCOPE LOOP AND HIT CONTINUE
3568
3569 015272 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3570 015274 012713 000326 MOV #326,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3571 015300 000000 HLT
3572 015302 022737 015263 021770 R7TR2: CMP #R7TR2A+3,BUFF-4
3573 015310 001405 BEQ 15 ;CORRECT PC NOT ON STACK
3574
3575 015312 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3576 015314 012713 000327 MOV #327,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3577 015320 000000 HLT
  
```

G06

MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 71
 DOKDBA.P11 17-FEB-77 08:26

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

```

3578 015322 000240          LOOP          ; REPLACE THIS INSTRUCTION BY A
3579                                     ; "JMP 2R0" (000110) TO GET A
3580                                     ; SCOPE LOOP AND HIT CONTINUE
3581 015324 010700          1S:          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    2*STESTN
3587 015332 012706 021774    MOV     #BUFF,%6
3588 015336 012737 015360 000004    MOV     #R7TR3,4
3589 015344 005307          DEC     %7          ; MAKE PC ODD
3590 015346          R7TR3A:
3591
3592 015346 005212          INC     (R2)        ; SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3593 015350 012713 000330    MOV     #330,(R3)  ; MOVE TO MAIL BOX ($FATAL) THE ERROR
3594 015354 000000          HLT
3595 015356 000240          LOOP          ; SHOULD TRAP
3596                                     ; REPLACE THIS INSTRUCTION BY A
3597                                     ; "JMP 2R0" (000110) TO GET A
3598 015360 022737 015347 021770  R7TR3:  CMP     #R7TR3A+1,BUFF-4 ; CHECK VALUE OF PC ON STACK
3599 015366 001405          BEQ     1S
3600
3601 015370 005212          INC     (R2)        ; SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3602 015372 012713 000331    MOV     #331,(R3)  ; MOVE TO MAIL BOX ($FATAL) THE ERROR
3603 015376 000000          HLT          ; WRONG VALUE ON STACK
3604 015400 000240          LOOP          ; REPLACE THIS INSTRUCTION BY A
3605                                     ; "JMP 2R0" (000110) TO GET A
3606                                     ; SCOPE LOOP AND HIT CONTINUE
3607
3608 015402 010700          1S:          SCOPE
3609                                     ; *****
3610                                     ; *TEST 132 TEST THAT "SEC, ROL PC" RESULTS IN TRAP
3611                                     ; *****
3612
3613 015404 005237 000304    TST132:  INC    2*STESTN
3614 015410 012706 021774    MOV     #BUFF,%6
3615 015414 012737 015450 000004    MOV     #R7TR4,4
3616 015422 000261          SEC
3617 015424 006107          ROL     %7          ; CARRY EQUALS A 1
3618 015426          TR4A:          ; PC BECOMES ODD
3619
3620 015426 005212          INC     (R2)        ; SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3621 015430 012713 000332    MOV     #332,(R3)  ; MOVE TO MAIL BOX ($FATAL) THE ERROR
3622 015434 000000          HLT
3623 015436 000240          LOOP          ; REPLACE THIS INSTRUCTION BY A
3624                                     ; "JMP 2R0" (000110) TO GET A
3625                                     ; SCOPE LOOP AND HIT CONTINUE
3626
3627 015440 005212          INC     (R2)        ; SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3628 015442 012713 000333    MOV     #333,(R3)  ; MOVE TO MAIL BOX ($FATAL) THE ERROR
3629 015446 000000          HLT
3630 015450 022737 033057 021770  R7TR4:  CMP     #TR4A+TR4A+3,BUFF-4 ; CHECK FOR VALUE ON STACK
3631 015456 001405          BEQ     1S
3632
3633 015460 005212          INC     (R2)        ; SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
  
```


H06

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 72
 DOKDBA.P11 17-FEB-77 08:26 T132 TEST THAT "SEC, ROL PC" RESULTS IN TRAP

```

3634 015462 012713 000334          MOV      #334, (R3)          ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3635 015466 000200          HLT                               ;WRONG VALUE ON STACK
3636 015470 000240          LOOP                            ;REPLACE THIS INSTRUCTION BY A
3637                                     ;"JMP 2R0" (000110) TO GET A
3638                                     ;SCOPE LOOP AND HIT CONTINUE
3639 015472 012737 000006 000004 15:  MOV      #6, 4              ;RESET UP A HALT FOR TRAP
3640
3641 015500 010700          SCOPE
3642
3643                                     ;*****
3644                                     ;*TEST 133 TEST THAT A PENDING TTY INTERRUPT OCCURS BEFOR A "TRAP" TRAP
3645                                     ;*****
3646 015502 005237 000304          TST133: INC      @#STESTN
3647 015506 012706 021774          MOV      #BUFF, %6
3648 015512 012737 000340 177776  MOV      #340, $STATUS      ;HIGHEST PRIORITY LEVEL
3649 015520 012737 000100 177564  MOV      #100, TTCSR        ;INTERRUPT FOR TTY PUNCH/PRINTER
3650 015526 012737 015564 000034  MOV      #TR1, 34          ;"TRAP" VECTOR
3651 015534 012737 015570 000064  MOV      #TR2, 64          ;TTY VECTOR
3652 015542 012737 000340 000036  MOV      #340, 36          ;IF TRAP TRAPS, MOVE 340 TO PRIORITY
3653 015550 005037 177776          CLR      STATUS            ;SHOULD TRAP AT IND OF CLR INST
3654 015554 104400          TRAP                          ;TTY INTERRUPT SHOULD OVERRIDE TRAP
3655
3656 015556 005212          INC      (R2)              ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3657 015560 012713 000335          MOV      #335, (R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3658 015564 000000          TR1:   HLT                               ;TRAP OCCUR FIRST
3659 015566 000240          LOOP                            ;REPLACE THIS INSTRUCTION BY A
3660                                     ;"JMP 2R0" (000110) TO GET A
3661                                     ;SCOPE LOOP AND HIT CONTINUE
3662 015570 005037 177564          TR2:   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 015602 005237 000304          TST134: INC      @#STESTN
3670 015606 012706 021774          MOV      #BUFF, %6
3671 015612 012737 000340 177776  MOV      #340, $STATUS      ;HIGHEST PRIORITY LEVEL
3672 015620 012737 000100 177564  MOV      #100, TTCSR        ;INTERRUPT FOR TTY PUNCH/PRINTER
3673 015626 012737 015664 000034  MOV      #TR3, 34          ;"TRAP" VECTOR
3674 015634 005037 000036          CLR      36                ;TTY VECTOR
3675 015640 012737 015700 000064  MOV      #TR4, 64          ;IF TRAP TRAPS, MOVE 340 TO PRIORITY
3676 015646 012737 015674 000020  MOV      #TR5, 20          ;SHOULD TRAP AT IND OF CLR INST
3677 015654 012737 000340 000022  MOV      #340, 22          ;TTY INTERRUPT SHOULD OVERRIDE TRAP
3678 015662 104400          TR3:   TRAP                          ;TRAP
3679 015664 000004          IOT                            ;TTY OUTPUT
3680                                     ;IOT
3681                                     ;IOT PRIORITY
3682                                     ;THE ACT OF TRAPPING LOWER PRIORITY
3683                                     ;INTERRUPT SHOULD OCCURE INPLACE OF IOT TRAP
3684 015666 005212          INC      (R2)              ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3685 015670 012713 000336          TR5:   MOV      #336, (R3)    ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3686 015674 000000          HLT                               ;NO INTERRUPT BETWEEN TRAPS
3687 015676 000240          LOOP                            ;REPLACE THIS INSTRUCTION BY A
3688                                     ;"JMP 2R0" (000110) TO GET A
3689                                     ;SCOPE LOOP AND HIT CONTINUE
3687 015700 005037 177564          TR4:   CLR      TTCSR
3688 015704 005037 000022          CLR      22                ;CLR IOT PRIORITY
3689 015710 010700          SCOPE
  
```

```

3690 ;*****
3691 ;*TEST 135 TEST THAT 'T' BIT TRAP OCCURS BEFORE AN INTERRUPT IS ACKNOWLEDGED
3692 ;*****
3693 015712 005237 000304 TST135: INC #STESTN
3694 015716 012737 000340 177776 MOV #340,CC ;SET PRIORITY =7
3695 015724 012737 000100 177564 MOV #100,TTCSR ;ENABLE INTERRUPT ON TTY PRINTER
3696 015732 012737 015770 000014 MOV #INT,TBITVEC ;LOAD 'T' BIT VECTOR
3697 015740 005037 000016 CLR TBITVEC+2
3698 015744 012737 016002 000064 MOV #INT2,64 ;LOAD TTY INT. VECTOR
3699 015752 012706 021774 MOV #BUFF,%6 ;SET STACK POINTER
3700 015756 012746 000020 MOV #20,-(6) ;PUSH 'T' BIT ON THE STACK
3701 015762 012746 015770 MOV #.+6,-(6) ;PUSH PC ON THE STACK
3702 015766 000002 RTI ;SET 'T' BIT
3703 015770 INT:
3704
3705 015770 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3706 015772 012713 000337 MOV #337,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3707 015776 000000 HLT ;ERROR! TTY INTERRUPTED
3708 016000 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
3709 ;"JMP 2R0" (000110) TO GET A
3710 ;SCOPE LOOP AND HIT CONTINUE
3711 016002 005037 177564 INT2: CLR TTCSR
3712 016006 022737 000020 021772 CMP #20,BUFF-2
3713 016014 001405 BEQ 15
3714
3715 016016 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3716 016020 012713 000340 MOV #340,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3717 016024 000000 HLT
3718 016026 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
3719 ;"JMP 2R0" (000110) TO GET A
3720 ;SCOPE LOOP AND HIT CONTINUE
3721 016030 022737 015770 021770 15: CMP #INT,BUFF-4
3722 016036 001405 BEQ 25
3723
3724 016040 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3725 016042 012713 000341 MOV #341,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3726 016046 000000 HLT
3727 016050 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
3728 ;"JMP 2R0" (000110) TO GET A
3729 ;SCOPE LOOP AND HIT CONTINUE
3730 016052 022737 000000 021766 25: CMP #0,BUFF-6
3731 016060 001405 BEQ 35
3732
3733 016062 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3734 016064 012713 000342 MOV #342,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3735 016070 000000 HLT
3736 016072 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
3737 ;"JMP 2R0" (000110) TO GET A
3738 ;SCOPE LOOP AND HIT CONTINUE
3739 016074 022737 015770 021764 35: CMP #INT,BUFF-10
3740 016102 001405 BEQ INT1
3741
3742 016104 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3743 016106 012713 000343 MOV #343,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3744 016112 000000 HLT
3745 016114 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
  
```

JOB

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 74
 DOKDBA.P11 17-FEB-77 08:26

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

```

3746                                     ;"JMP 2R0" (000110) TO GET A
3747                                     ;SCOPE LOOP AND HIT CONTINUE
3748 016116 010700          INT1:  SCOPE
3749
3750                                     ;*****
3751                                     ;*TEST 136 TEST THAT "RESET" GOES TO OUTSIDE WORLD
3752                                     ;*****
3753 016120 005237 000304          TST136: INC  2*STESTN
3754 016124 012706 021774          MOV   #BUFF,%6
3755 016130 012737 000340 177776  MOV   #340,CC          ;LOCK OUT INTERRUPTS
3756 016136 012737 016206 000064  MOV   #TR6,64          ;LOAD TELEPRINTER VECTOR
3757 016144 012737 000100 177564  MOV   #100,TTCSR       ;SET INTERRUPT ENABLE
3758 016152 000005                                RESET  ;SHOULD CLEAR INTERRUPT ENABLE
3759 016154 032737 000100 177564  BIT   #100,TTCSR       ;TEST FOR CLEAR
3760 016162 001405                                BEQ   IS
3761
3762 016164 005212                                INC   (R2)              ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3763 016166 012713 000344          MOV   #344,(R3)         ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3764 016172 000000                                HLT
3765 016174 000240          LOOP
3766                                     ;REPLACE THIS INSTRUCTION BY A
3767                                     ;"JMP 2R0" (000110) TO GET A
3768 016176 000406          IS:      BR    TR6X              ;SCOPE LOOP AND HIT CONTINUE
3769                                     ;GO TO SCOPE
3770 016200 005212                                INC   (R2)              ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3771 016202 012713 000345          TR6:  MOV   #345,(R3)         ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3772 016206 000000          HLT
3773                                     ;ERROR! TELETYPE INTERRUPTED WHEN
3774 016210 000240          LOOP
3775                                     ;PROCESSOR WAS AT LEVEL 7
3776                                     ;REPLACE THIS INSTRUCTION BY A
3777 016212 022626                                CMP   (6)+,(6)+
3778 016214 005037 177564          TR6X: 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  2*STESTN
3785 016226 012706 021774          MOV   #BUFF,%6          ;SET STACK
3786 016232 005037 177776          CLR   STATUS            ;ALLOW INTERRUPT
3787 016236 012737 016252 000064  MOV   #RESET1,64        ;TTY INTERRUPT VECTOR
3788 016244 052737 000100 177564  BIS   #100,TTCSR       ;SET INTERRUPT ENABLE
3789 016252 000005          RESET1: RESET        ;IF THIS HANGS CHECK SACK
3790 016254 012737 000066 000064  MOV   #66,64           ;FOR FALSE INTERRUPT
3791 016262 010700          SCOPE
3792
3793                                     ;*****
3794                                     ;*TEST 140 TEST RESET WITH TRACE ON
3795                                     ;*****
3795 016264 005237 000304          TST140: INC  2*STESTN
3796 016270 012706 021774          MOV   #BUFF,%6          ;SET STACK
3797 016274 012737 016336 000014  MOV   #RESET2,TBITVEC ;SET UP TRACE VECTOR
3798 016302 005037 000016          CLR   TBITVEC+2
3799 016306 012746 000020          MOV   #20,-(6)         ;PUSH 'T' BIT ON THE STACK
3800 016312 012746 016320          MOV   #.+6,-(6)       ;PUSH PC ON THE STACK
3801 016316 000002          RTI                    ;SET 'T' BIT

```

```

3802 016320 007005 RESET ; SHOULD HAVE NO EFFECT
3803 016322 007005 RESET ; NO EFFECT
3804
3805 016324 005212 INC (R2) ; SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3806 016326 012713 000346 MOV #346, (R3) ; MOVE TO MAIL BOX ($FATAL) THE ERROR
3807 016332 000000 HLT ; TRACE TRAP FAILED
3808 016334 000240 LOOP ; REPLACE THIS INSTRUCTION BY A
3809 ; "JMP 2RD" (000110) TO GET A
3810 ; SCOPE LOOP AND HIT CONTINUE
3811 016336 005037 177776 RESET2: CLR STATUS ; CLEAR TRACK
3812 016342 005037 000016 CLR 16 ; TRACE STATUS
3813 016346 010700 SCOPE
3814
3815 ; *****
3816 ; *TEST 141 TEST THAT WHEN TTY INTERRUPTS IT POPS NEW STATUS
3817 ; *****
3818 016350 005237 000304 TST141: INC #TESTN
3819 016354 000005 RESET
3820 016356 012706 021774 MOV #BUFF,%6 ; SET UP STACK
3821 016362 012737 016406 000064 MOV #TTY3,64 ; INTERRUPT VECTOR
3822 016370 005037 177776 CLR STATUS ; DROP PROCESSOR PRIORITY
3823 016374 012737 000357 000066 MOV #357,66 ; 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 #357,-2
3829 016422 001405 BEQ 15
3830
3831 016424 005212 INC (R2) ; SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3832 016426 012713 000347 MOV #347, (R3) ; MOVE TO MAIL BOX ($FATAL) THE ERROR
3833 016432 000000 HLT ; INTERRUPT DID NOT POP CORRECT STATUS
3834 016434 000240 LOOP ; REPLACE THIS INSTRUCTION BY A
3835 ; "JMP 2RD" (000110) TO GET A
3836 ; SCOPE LOOP AND HIT CONTINUE
3837 016436 000005 15: RESET ; CLR INTERRUPT ENABLE
3838 016440 010700 SCOPE
3839 016442 012706 021774 MOV #BUFF,%6 ; STACK SET UP
3840 016446 012737 016472 000064 MOV #TTY4,64 ; INTERRUPT VECTOR
3841 016454 005037 000066 CLR 66 ; CLR NEW STATUS
3842 016460 012737 000157 177776 MOV #157,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 BEQ 15
3848
3849 016506 005212 INC (R2) ; SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3850 016510 012713 000350 MOV #350, (R3) ; MOVE TO MAIL BOX ($FATAL) THE ERROR
3851 016514 000000 HLT ; INTERRUPT DID NOT POP CORRECT STATUS
3852 016516 000240 LOOP ; REPLACE THIS INSTRUCTION BY A
3853 ; "JMP 2RD" (000110) TO GET A
3854 ; SCOPE LOOP AND HIT CONTINUE
3855 016520 005037 177564 15: CLR TTCSR
3856 016524 010700 SCOPE
3857 ; *****

```

L06

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 76
 DOKDBA.P11 17-FEB-77 08:26 T142 CONTIGUOUS MEMORY ADDRESS TEST

```

3858                                     ;*TEST 142 CONTIGUOUS MEMORY ADDRESS TEST
3 59                                     ;*****
3 60 016526 005237 000304 TST142: INC @#STESTN
3 61                                     ;THIS TEST CHECKS THAT ALL MEMORY (UP TO 28K) IS CONTIGUOUS
3 62 016532 012706 021774 MOV @#BUFF,%6 ;SET STACK POINTER
3 63 016536 012737 016552 000004 MOV @#MEMEND,@#4 ;SET TIME OUT TRAP VECTOR
3864 016544 005000 CLR %0 ;SET STARTING ADDRESS FOR TEST
3865 016546 005720 TST (0)+ ;BEGIN
3866 016550 000776 BR -2 ;LOOP UNTIL TIMEOUT OCCURS
3867 016552 022626 MEMEND: CMP (6)+,(6)+ ;RESTORE STACK POINTER
3868 016554 022700 160002 CMP @#160002,%0 ;AT END OF 28K MEMORY??
3869 016560 001406 BEQ MEMEX ;YES - EXIT TEST
3870 016562 005720 TST (0)+ ;SHOULD TIME OUT HERE UP TO 28K
3871
3872 016564 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3873 016566 012713 000351 MOV @#351,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3874 016572 000000 HLT ;ERROR - FAILED TO TIME OUT!
3875 016574 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
3876 ;"JMP @#0" (000110) TO GET A
3877 ;SCOPE LOOP AND HIT CONTINUE
3878 016576 012737 000006 000004 MEMEX: MOV @#6,@#4 ;RESTORE TRAPCATCHER
3879 016604 005037 000006 CLR @#6
3880 016610 010700 SCOPE
  
```

MO6

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 77
 DOKDBA.P11 17-FEB-77 08:26

T142 CONTIGUOUS MEMORY ADDRESS TEST

;SPECIAL CASE OF ODD;.EVEN .BYTE AND REGISTER 6

```

3881
3882
3883 ;*****
3884 ;*TEST 143 TEST AUTO INCREMENT & DECREMENT OF R6 - WORD & BYTE
3885 ;*****
3886 016612 005237 000304 TST143: INC @#STESTN
3887
3888 016616 012706 021774 AIDR6: MOV #BUFF,%6 ;SET UP THE STACK
3889 016622 112627 MOV (6)+,(PC)+ ;SIX SHOULD INCREMENT BY TWO
3890 016624 000000 .WORD 0
3891 016626 020627 021776 CMP %6,#BUFF+2
3892 016632 001405 BEQ 1$
3893
3894 016634 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3895 016636 012713 000352 MOV #352,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3896 016642 000000 HLT ;R6 DID NOT AUTO INCREMENT BY TWO
3897 016644 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
3898 ;"JMP @RO" (000110) TO GET A
3899 ;SCOPE LOOP AND HIT CONTINUE
3900 016646 010700 1$: SCOPE
3901
3902 016650 012706 001000 MOV #1000,%6
3903 016654 114627 MOV (6)+,(PC)+ ;SHOULD DECREMENT BY TWO
3904 016656 000000 .WORD 0
3905 016660 020627 000776 CMP %6,#776
3906 016664 001405 BEQ 2$
3907
3908 016666 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3909 016670 012713 000353 MOV #353,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3910 016674 000000 HLT ;R6 DID NOT AUTO DECREMENT BY 2
3911 016676 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
3912 ;"JMP @RO" (000110) TO GET A
3913 ;SCOPE LOOP AND HIT CONTINUE
3914 016700 010700 2$: SCOPE
3915
3916 016702 012706 021774 MOV #BUFF,%6
3917 016706 112626 MOV (6)+,(6)+ ;DOUBLE AUTO INCREMENT OF R6
3918 016710 020627 022000 CMP %6,#BUFF+4
3919 016714 001405 BEQ 3$
3920
3921 016716 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3922 016720 012713 000354 MOV #354,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3923 016724 000000 HLT ;WRONG AUTO INCREMENT OF R6
3924 016726 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
3925 ;"JMP @RO" (000110) TO GET A
3926 ;SCOPE LOOP AND HIT CONTINUE
3927 016730 010700 3$: SCOPE
3928
3929 016732 012706 021774 MOV #BUFF,%6
3930 016736 005004 CLR %4
3931 016740 122624 CMPB (6)+,(4)+ ;TEST INCREMENT OF R6
3932 016742 020627 021776 CMP %6,#BUFF+2
3933 016746 001405 BEQ 4$
3934
3935 016750 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3936 016752 012713 000355 MOV #355,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
  
```

NO6

.MAIN. *ACY11 2/(1006) 17-FEB-77 08:32 PAGE 78
 DOKDBA.P11 17-FEB-77 08:26

T143 TEST AUTG INCREMENT & DECREMENT OF R6 - WORD & BYTE

3937	016755	000000		HLT		;WRONG INCREMENT OF R6
3938	016760	000240		LOOP		;REPLACE THIS INSTRUCTION BY A
3939						; "JMP 2RD" (000110) TO GET A
3940						;SCOPE LOOP AND HIT CONTINUE
3941	016762	010700		4S: SCOPE		
3942						
3943	016764	012706	021774	MOV	#BUFF,%6	
3944	016770	005004		CLR	%4	
3945	016772	122426		CMPB	(4)+,(6)+	;TEST INCREMENT OF R6
3946	016774	020627	021776	CMP	%6,#BUFF+2	
3947	017000	001405		BEQ	5S	
3948						
3949	017002	005212		INC	(R2)	;SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
3950	017004	012713	000356	MOV	#356,(R3)	;MOVE TO MAIL BOX (\$FATAL) THE ERROR
3951	017010	070000		HLT		;WRONG INCREMENT OF R6
3952	017012	000240		LOOP		;REPLACE THIS INSTRUCTION BY A
3953						; "JMP 2RD" (000110) TO GET A
3954						;SCOPE LOOP AND HIT CONTINUE
3955	017014	010700		5S: SCOPE		
3956						
3957	017016	012706	021774	MOV	#BUFF,%6	
3958	017022	005004		CLR	%4	
3959	017024	122624		CMPB	(6)+,(4)+	;TEST INCREMENT OF R4
3960	017026	020427	000001	CMP	%4,#1	
3961	017032	001405		BEQ	6S	
3962						
3963	017034	005212		INC	(R2)	;SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
3964	017036	012713	000357	MOV	#357,(R3)	;MOVE TO MAIL BOX (\$FATAL) THE ERROR
3965	017042	000000		HLT		;WRONG INCREMENT OF R4
3966	017044	000240		LOOP		;REPLACE THIS INSTRUCTION BY A
3967						; "JMP 2RD" (000110) TO GET A
3968						;SCOPE LOOP AND HIT CONTINUE
3969	017046	010700		6S: SCOPE		
3970	017050	012706	021774	MOV	#BUFF,%6	
3971	017054	005004		CLR	%4	
3972	017056	122446		CMPB	(4)+,-(6)	;TEST DECREMENT OF R6
3973	017060	020627	021772	CMP	%6,#BUFF-2	
3974	017064	001405		BEQ	7S	
3975						
3976	017066	005212		INC	(R2)	;SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
3977	017070	012713	000360	MOV	#360,(R3)	;MOVE TO MAIL BOX (\$FATAL) THE ERROR
3978	017074	000000		HLT		;WRONG INCREMENT OF R6
3979	017076	000240		LOOP		;REPLACE THIS INSTRUCTION BY A
3980						; "JMP 2RD" (000110) TO GET A
3981						;SCOPE LOOP AND HIT CONTINUE
3982	017100	010700		7S: SCOPE		
3983						
3984	017102	012706	021774	MOV	#BUFF,%6	
3985	017106	005004		CLR	%4	
3986	017110	122426		CMPB	(4)+,(6)+	;TEST INCREMENT OF R4
3987	017112	020427	000001	CMP	%4,#1	
3988	017116	001405		BEQ	8S	
3989						
3990	017120	005212		INC	(R2)	;SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
3991	017122	012713	000361	MOV	#361,(R3)	;MOVE TO MAIL BOX (\$FATAL) THE ERROR
3992	017126	000000		HLT		;WRONG INCREMENT OF R4

3993	017130	000240		LOOP						; REPLACE THIS INSTRUCTION BY A ; "JMP 2R0" (000110) TO GET A ; SCOPE LOOP AND HIT CONTINUE
3994										
3995										
3996	017132	010700		8\$:	SCOPE					
3997										
3998	017134	012706	001000		MOV	#1000,%6				
3999	017140	124627			CMPB	-(6),(PC)+				; TEST DECREMENT OF R6
4000	017142	022706	000776		CMP	#776,%6				
4001	017146	001405			BEQ	9\$				
4002										
4003	017150	005212			INC	(R2)				; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
4004	017152	012713	000362		MOV	#362,(R3)				; MOVE TO MAIL BOX (\$FATAL) THE ERROR
4005	017156	000000			HLT					; WRONG DECREMENT OF R6
4006	017160	000240			LOOP					; REPLACE THIS INSTRUCTION BY A ; "JMP 2R0" (000110) TO GET A ; SCOPE LOOP AND HIT CONTINUE
4007										
4008										
4009	017162	010700		9\$:	SCOPE					
4010										
4011	017164	010700			SCOPE					; 5 JAN 71
4012	017166	012706	021774		MOV	#BUFF,%6				; INITIAL POSITION OF R6
4013	017172	012737	177777 021772		MOV	#-1,BUFF-2				; HIGH BYTE WILL BE DATA
4014	017200	012737	025252 001000		MOV	#25252,K1				
4015	017206	114637	001001		MOVW	-(6),K1+1				; MOV TO ODD ADDRESS
4016	017212	022737	177652 001000		CMP	#177652,K1				; DID MOVW WORK CORRECTLY
4017	017220	001405			BEQ	10\$				
4018										
4019	017222	005212			INC	(R2)				; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
4020	017224	012713	000363		MOV	#363,(R3)				; MOVE TO MAIL BOX (\$FATAL) THE ERROR
4021	017230	000000			HLT					; MOVW -(6), ODD: FAILED
4022	017232	000240			LOOP					; REPLACE THIS INSTRUCTION BY A ; "JMP 2R0" (000110) TO GET A ; SCOPE LOOP AND HIT CONTINUE
4023										
4024										
4025	017234	010700		10\$:	SCOPE					
4026										
4027	017236	012706	021774		MOV	#BUFF,%6				; INITIAL POSITION OF R6
4028	017242	012737	177777 021772		MOV	#-1,BUFF-2				; HIGH BYTE SOURCE DATA
4029	017250	012737	125252 001000		MOV	#125252,K1				; INITIAL SET UP OF DESTINATION
4030	017256	114637	001000		MOVW	-(6),K1				; HIGH BYTE OF BUFF-2, TO LOW BYTE OF K1
4031	017262	022737	125377 001000		CMP	#125377,K1				; TEST RESULTS
4032	017270	001405			BEQ	11\$				
4033										
4034	017272	005212			INC	(R2)				; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
4035	017274	012713	000364		MOV	#364,(R3)				; MOVE TO MAIL BOX (\$FATAL) THE ERROR
4036	017300	000000			HLT					; MOVW -(6), EVEN, FAILED
4037	017302	000240			LOOP					; REPLACE THIS INSTRUCTION BY A ; "JMP 2R0" (000110) TO GET A ; SCOPE LOOP AND HIT CONTINUE
4038										
4039										
4040	017304	010700		11\$:	SCOPE					


```

4041                                     ;*****
4042                                     ;TEST 144 TEST TRANSFER OF .BYTE USING R6
4043                                     ;*****
4044 017306 005237 000304 TST144: INC @#STESTN
4045 017312 012737 123456 001010 BXFR: MOV #123456,K5
4046 017320 012737 050505 001000 MOV #050505,K1
4047 017326 012705 001000 MOV #K1,%5 ;%5=(050505)K1
4048 017332 012706 001010 MOV #K5,%6 ;%6(123456)K5
4049 017336 112625 MOVB (6)↑,(5)+ ;LOW .BYTE OF R6 TO R5
4050 017340 022737 050456 001000 CMP #050456,K1
4051 017346 001405 BEQ 1$
4052
4053 017350 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4054 017352 012713 000365 MOV #365,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4055 017356 000000 HLT ;FALSE TRANSFER OF .BYTE
4056 017360 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
4057 ;"JMP 2R0" (000110) TO GET A
4058 ;SCOPE LOOP AND HIT CONTINUE
4059 017362 010700 1$: SCOPE
4060
4061 017364 012737 123456 001010 MOV #123456,K5
4062 017372 012737 050505 001000 MOV #050505,K1
4063 017400 012705 001000 MOV #K1,%5 ;%5(050505)K1
4064 017404 012706 001012 MOV #K6,%6 ;%6(123456)K5
4065 017410 114625 MOVB -(6),(5)+ ;LOW .BYTE OF R6 TO R5 (DECREMENT)
4066 017412 023727 001000 050456 CMP K1,#050456
4067 017420 001405 BEQ 2$
4068
4069 017422 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4070 017424 012713 000366 MOV #366,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4071 017430 000000 HLT ;FALSE R6 .BYTE TRANSFER
4072 017432 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
4073 ;"JMP 2R0" (000110) TO GET A
4074 ;SCOPE LOOP AND HIT CONTINUE
4075 017434 010700 2$: SCOPE
4076
4077 017436 012737 123456 001000 MOV #123456,K1
4078 017444 012737 050505 001010 MOV #050505,K5
4079 017452 012705 001000 MOV #K1,%5 ;(123456)
4080 017456 012706 001010 MOV #K5,%6 ;(050505)
4081 017462 112526 MOVB (5)↑,(6)+ ;LOW OF R5 TO LOW OF R6
4082 017464 022737 050456 001010 CMP #050456,K5
4083 017472 001405 BEQ 3$
4084
4085 017474 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4086 017476 012713 000367 MOV #367,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4087 017502 000000 HLT ;FALSE R6 .BYTE TRANSFER
4088 017504 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
4089 ;"JMP 2R0" (000110) TO GET A
4090 ;SCOPE LOOP AND HIT CONTINUE
4091 017506 010700 3$: SCOPE
4092
4093 017510 012737 123456 001000 MOV #123456,K1
4094 017516 012737 050505 001010 MOV #050505,K5
4095 017524 012705 001001 MOV #K1+1,%5 ;123456
4096 017530 012706 001010 MOV #K5,%6 ;050505

```

4097	017534	112526				MOVB	(5)+,(6)+	;HIGH OF R5 TO LOW OF R6
4098	017536	023727	001010	050647		CMP	K5,#050647	
4099	017544	001405				BEQ	4\$	
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		;FALSE R6 .BYTE TRANSFER
4104	017556	000240				LOOP		;REPLACE THIS INSTRUCTION BY A
4105								"JMP 2RD" (000110) TO GET A
4106								;SCOPE LOOP AND HIT CONTINUE
4107	017560	010700			4\$:	SCOPE		
4108								
4109	017562	012737	123456	001000		MOV	#123456,K1	
4110	017570	012737	050505	001010		MOV	#050505,K5	
4111	017576	012705	001001			MOV	#K1+1,%5	;R5-123456--000 ADDRESS
4112	017602	012706	001010			MOV	#K5,%6	;R6-050505--.EVEN ADDRESS
4113	017606	112625				MOVB	(6)+,(5)+	;LOW OF R6 TO HIGH OF R5
4114	017610	022737	042456	001000		CMP	#042456,K1	
4115	017616	001405				BEQ	5\$	
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		;FAILED LOW OF 6 TO HIGH OF 5
4120	017630	000240				LOOP		;REPLACE THIS INSTRUCTION BY A
4121								"JMP 2RD" (000110) TO GET A
4122								;SCOPE LOOP AND HIT CONTINUE
4123	017632	010700			5\$:	SCOPE		
4124								
4125								;*****
4126								;*TEST 145 TEST .BYTE OPERATION WITH SEQUENTIAL ODD;.EVEN ADDRESS
4127								;*****
4128	017634	005237	000304		TST145:	INC	2#STESTN	
4129	017640	123737	001014	001015	BYTOP:	CMPB	K7,K7+1	;SAME .WORD LOW TO HIGH
4130	017646	001405				BEQ	1\$	
4131								
4132	017650	005212				INC	(R2)	;SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
4133	017652	012713	000372			MOV	#372,(R3)	;MOVE TO MAIL BOX (\$FATAL) THE ERROR
4134	017656	000000				HLT		;SHOULD COMPARE LOW TO HIGH
4135	017660	000240				LOOP		;REPLACE THIS INSTRUCTION BY A
4136								"JMP 2RD" (000110) TO GET A
4137								;SCOPE LOOP AND HIT CONTINUE
4138	017662	010700			1\$:	SCOPE		
4139								
4140	017664	123737	001015	001014		CMPB	K7+1,K7	;COMPARE ODD TO .EVEN SAME .WORD
4141	017672	001405				BEQ	2\$	
4142								
4143	017674	005212				INC	(R2)	;SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
4144	017676	012713	000373			MOV	#373,(R3)	;MOVE TO MAIL BOX (\$FATAL) THE ERROR
4145	017702	000000				HLT		;ODD TO .EVEN .BYTE FAILURE
4146	017704	000240				LOOP		;REPLACE THIS INSTRUCTION BY A
4147								"JMP 2RD" (000110) TO GET A
4148								;SCOPE LOOP AND HIT CONTINUE
4149	017706	010700			2\$:	SCOPE		
4150								
4151	017710	123737	001017	001014		CMPB	K10+1,K7	;SEQUENTIAL .BYTES
4152	017716	001405				BEQ	3\$;DIFFERENT .WORDS

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 83
DOKDBA.P11 17-FEB-77 08:26

T145 TEST .BYTE OPERATION WITH SEQUENTIAL ODD;.EVEN ADDRESS

4209	020064	005212		INC	(R2)	; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
4210	020066	012713	000401	MOV	#401,(R3)	; MOVE TO MAIL BOX (\$FATAL) THE ERROR
4211	020072	000000		HLT		; EVEN TO ODD FAILED
4212	020074	000240		LOOP		; REPLACE THIS INSTRUCTION BY A
4213						; "JMP JRO" (000110) TO GET A
4214						; SCOPE LOOP AND HIT CONTINUE
4215	020076	010700		85:	SCOPE	
4216						

```

4217
4218
4219
4220
4221 020100 005237 000304 TST146: INC @#STESTN
4222 020104 012737 177777 001020 RLOBYT: MOV #-1,K11
4223 020112 012700 000125 MOV #125,%0 ;LOAD R0
4224 020116 110037 001020 MOVB %0,K11
4225 020122 023727 001020 177525 CMP K11,#177525 ;WAS ONLY LSH MOVED?
4226 020130 001405 BEQ 1$
4227
4228 020132 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4229 020134 012713 000402 MOV #402,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4230 020140 000000 HLT ;ERROR! MOVB %R FAILED
4231 020142 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
4232 ;"JMP @RO" (000110) TO GET A
4233 ;SCOPE LOOP AND HIT CONTINUE
4234 020144 010700 1$: SCOPE
4235
4236 020146 012700 001020 MOV #K11,%0
4237 020152 010037 001020 MOV %0,K11
Z 4238 020156 110020 MOVB %0,(0)+
4239 020160 022737 001021 001020 CMP #K11+1,K11
4240 020166 001405 BEQ 2$
4241
4242 020170 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4243 020172 012713 000403 MOV #403,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4244 020176 000000 HLT
4245 020200 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
4246 ;"JMP @RO" (000110) TO GET A
4247 ;SCOPE LOOP AND HIT CONTINUE
4248 020202 010700 2$: SCOPE
4249
4250 020204 012706 001020 MOV #K11,%6
Z 4251 020210 010637 001020 MOV %6,K11
4252 020214 110626 MOVB %6,(6)+
4253 020216 023727 001020 001022 CMP K11,#K11+2
4254 020224 001405 BEQ 3$
4255
4256 020226 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4257 020230 012713 000404 MOV #404,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4258 020234 000000 HLT ;FAILED MOVB %6,(6)+
4259 020236 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
4260 ;"JMP @RO" (000110) TO GET A
4261 ;SCOPE LOOP AND HIT CONTINUE
4262 020240 010700 3$: SCOPE
4263
Z 4264 020242 012706 001020 MOV #K11,%6
4265 020246 010626 MOV %6,(6)+
4266 020250 023727 001020 001022 CMP K11,#K11+2
4267 020256 001405 BEQ 4$
4268
4269 020260 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4270 020262 012713 000405 MOV #405,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4271 020266 000000 HLT ;FAILED MOV %6,(6)+
4272 020270 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
  
```

H07

MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 85
 DOKDBA.P11 17-FEB-77 08:26

T146 TEST THAT MOV B %R MOVES ONLY THE LSH OF THE REGISTER

4273											; "JMP 2R0" (000110) TO GET A
4274											; SCOPE LOOP AND HIT CONTINUE
4275	020272	010700		45:	SCOPE						
4276											
4277	020274	000277			SCC						; SET STATUS
4278	020276	005037	177776		CLR	STATUS					; CLEAR STATUS
4279	020302	103005			BCC	55					
4280											
4281	020304	005212			INC	(R2)					; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
4282	020306	012713	000406		MOV	#406,(R3)					; MOVE TO MAIL BOX (\$FATAL) THE ERROR
4283	020312	000000			HLT						; C NOT CLEAR
4284	020314	000240			LOOP						; REPLACE THIS INSTRUCTION BY A
4285											; "JMP 2R0" (000110) TO GET A
4286											; SCOPE LOOP AND HIT CONTINUE
4287	020316	102005		55:	BVC	65					
4288											
4289	020320	005212			INC	(R2)					; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
4290	020322	012713	000407		MOV	#407,(R3)					; MOVE TO MAIL BOX (\$FATAL) THE ERROR
4291	020326	000000			HLT						; V NOT CLEAR
4292	020330	000240			LOOP						; REPLACE THIS INSTRUCTION BY A
4293											; "JMP 2R0" (000110) TO GET A
4294											; SCOPE LOOP AND HIT CONTINUE
4295	020332	001005		65:	BNE	75					
4296											
4297	020334	005212			INC	(R2)					; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
4298	020336	012713	000410		MOV	#410,(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 2R0" (000110) TO GET A
4302											; SCOPE LOOP AND HIT CONTINUE
4303	020346	100005		75:	BPL	85					
4304											
4305	020350	005212			INC	(R2)					; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
4306	020352	012713	000411		MOV	#411,(R3)					; MOVE TO MAIL BOX (\$FATAL) THE ERROR
4307	020356	000000			HLT						; N NOT CLEAR
4308	020360	000240			LOOP						; REPLACE THIS INSTRUCTION BY A
4309											; "JMP 2R0" (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	BIS	#17,STATUS					; SET STATUS TO ONES
4315	020374	103405			BCS	95					
4316											
4317											
4318	020376	005212			INC	(R2)					; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
4319	020400	012713	000412		MOV	#412,(R3)					; MOVE TO MAIL BOX (\$FATAL) THE ERROR
4320	020404	000000			HLT						; C NOT SET
4321	020406	000240			LOOP						; REPLACE THIS INSTRUCTION BY A
4322											; "JMP 2R0" (000110) TO GET A
4323											; SCOPE LOOP AND HIT CONTINUE
4324	020410	102405		95:	BVS	105					
4325											
4326	020412	005212			INC	(R2)					; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
4327	020414	012713	000413		MOV	#413,(R3)					; MOVE TO MAIL BOX (\$FATAL) THE ERROR
4328	020420	000000			HLT						; V NOT SET

4329	020422	000240			LOOP				; REPLACE THIS INSTRUCTION BY A
4330									; "JMP 2R0" (000110) TO GET A
4331									; SCOPE LOOP AND HIT CONTINUE
4332	020424	001405			10S: BEQ	11S			
4333									
4334	020426	005212			INC	(R2)			; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
4335	020430	012713	000414		MOV	#414, (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 2R0" (000110) TO GET A
4339									; SCOPE LOOP AND HIT CONTINUE
4340	020440	100405			11S: BMI	12S			
4341									
4342	020442	005212			INC	(R2)			; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
4343	020444	012713	000415		MOV	#415, (R3)			; MOVE TO MAIL BOX (\$FATAL) THE ERROR
4344	020450	000000			HLT				; N NOT SET
4345	020452	000240			LOOP				; REPLACE THIS INSTRUCTION BY A
4346									; "JMP 2R0" (000110) TO GET A
4347									; SCOPE LOOP AND HIT CONTINUE
4348	020454	010700			12S: SCOPE				
4349									
4350									
4351									; TEST THAT (%6) +4 TO %6
4352	020456	012706	021774		MOV	#BUFF, %6			; SET UP BUFFER POINTER
4353	020462	005037	021776		CLR	BUFF+2			; SET UP NEW STATUS
4354	020466	012737	020476	021774	MOV	#.+10, BUFF			; SET UP, RETURN
4355	020474	000006			RTT				; RETURN NEXT INSTRUCTION
4356	020476	020627	022000		CMP	%6, #BUFF+4			; %6 SHOULD BE PLUS 4
4357	020502	001405			BEQ	13S			
4358									
4359	020504	005212			INC	(R2)			; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
4360	020506	012713	000416		MOV	#416, (R3)			; MOVE TO MAIL BOX (\$FATAL) THE ERROR
4361	020512	000000			HLT				; %6 NOT INCREMENTED
4362	020514	000240			LOOP				; REPLACE THIS INSTRUCTION BY A
4363									; "JMP 2R0" (000110) TO GET A
4364									; SCOPE LOOP AND HIT CONTINUE
4365									
4366	020516	010700			13S: SCOPE				; TEST ((%6+2)) TO STATUS
4367	020520	012737	000357	021776	MOV	#357, BUFF+2			; RETURN STATUS ALL ONES
4368	020526	012706	021774		MOV	#BUFF, %6			; SET UP BUFFER POINTER
4369	020532	012737	020544	021774	MOV	#.+12, BUFF			; SET UP RETURN POINTER
4370	020540	000257			CCC				; CLEAR CONDITION CODES
4371	020542	000006			RTT				; RETURN
4372									
4373	020544	001405			BEQ	14S			
4374									
4375	020546	005212			INC	(R2)			; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
4376	020550	012713	000417		MOV	#417, (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 2R0" (000110) TO GET A
4380									; SCOPE LOOP AND HIT CONTINUE
4381	020560	102405			14S: BVS	15S			
4382									
4383	020562	005212			INC	(R2)			; SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
4384	020564	012713	000420		MOV	#420, (R3)			; MOVE TO MAIL BOX (\$FATAL) THE ERROR

J07

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 87
 DOKDBA.P11 17-FEB-77 08:26

T146 TEST THAT MOVB %R MOVES ONLY THE LSH OF THE REGISTER

4385	020570	000000			HLT				:V NOT SET
4386	020572	000240			LOOP				:REPLACE THIS INSTRUCTION BY A
4387									: "JMP 2R0" (000110) TO GET A
4388									: SCOPE LOOP AND HIT CONTINUE
4389	020574	100405			15\$: BMI	16\$			
4390									
4391	020576	005212			INC	(R2)			: SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
4392	020600	012713	000421		MOV	#421, (R3)			: MOVE TO MAIL BOX (\$FATAL) THE ERROR
4393	020604	000000			HLT				: N NOT SET
4394	020606	000240			LOOP				: REPLACE THIS INSTRUCTION BY A
4395									: "JMP 2R0" (000110) TO GET A
4396									: SCOPE LOOP AND HIT CONTINUE
4397	020610	103405			16\$: BCS	17\$			
4398									
4399	020612	005212			INC	(R2)			: SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
4400	020614	012713	000422		MOV	#422, (R3)			: MOVE TO MAIL BOX (\$FATAL) THE ERROR
4401	020620	000000			HLT				: C NOT SET
4402	020622	000240			LOOP				: REPLACE THIS INSTRUCTION BY A
4403									: "JMP 2R0" (000110) TO GET A
4404									: SCOPE LOOP AND HIT CONTINUE
4405									
4406	020624	010700			17\$: SCOPE				
4407	020626	005037	021776		CLR	BUFF+2			: ZERO TO STATUS
4408	020632	012706	021774		MOV	#BUFF,%6			
4409	020636	012737	020650	021774	MOV	#.+12,BUFF			
4410	020644	000277			SCC				: SET CONDITION CODES
4411	020646	000006			RTT				: RETURN
4412	020650	001005			BNE	18\$			
4413									
4414	020652	005212			INC	(R2)			: SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
4415	020654	012713	000423		MOV	#423, (R3)			: MOVE TO MAIL BOX (\$FATAL) THE ERROR
4416	020660	000000			HLT				: Z NOT CLEARED
4417	020662	000240			LOOP				: REPLACE THIS INSTRUCTION BY A
4418									: "JMP 2R0" (000110) TO GET A
4419									: SCOPE LOOP AND HIT CONTINUE
4420	020664	102005			18\$: BVC	19\$			
4421									
4422	020666	005212			INC	(R2)			: SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
4423	020670	012713	000424		MOV	#424, (R3)			: MOVE TO MAIL BOX (\$FATAL) THE ERROR
4424	020674	000000			HLT				: V NOT CLEARED
4425	020676	000240			LOOP				: REPLACE THIS INSTRUCTION BY A
4426									: "JMP 2R0" (000110) TO GET A
4427									: SCOPE LOOP AND HIT CONTINUE
4428	020700	100005			19\$: BPL	20\$			
4429									
4430	020702	005212			INC	(R2)			: SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
4431	020704	012713	000425		MOV	#425, (R3)			: MOVE TO MAIL BOX (\$FATAL) THE ERROR
4432	020710	000000			HLT				: N NOT CLEARED
4433	020712	000240			LOOP				: REPLACE THIS INSTRUCTION BY A
4434									: "JMP 2R0" (000110) TO GET A
4435									: SCOPE LOOP AND HIT CONTINUE
4436	020714	103005			20\$: BCC	21\$			
4437									
4438	020716	005212			INC	(R2)			: SET MESSAGE TYPE (\$MSGTY) TO FATAL ERROR
4439	020720	012713	000426		MOV	#426, (R3)			: MOVE TO MAIL BOX (\$FATAL) THE ERROR
4440	020724	000000			HLT				: C NOT CLEARED

K07

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 88
DOKD8A.P11 17-FEB-77 08:26

T146 TEST THAT MOVB %R MOVES ONLY THE LSH OF THE REGISTER

4441 020726 000240
4442
4443
4444 020730 010700
4445 020732

LOOP

;REPLACE THIS INSTRUCTION BY A
; "JMP PRO" (000110) TO GET A
; SCOPE LOOP AND HIT CONTINUE

215: SCOPE
RLOBY1:

T146 TEST THAT MOV B %R MOVES ONLY THE LSH OF THE REGISTER

```

4446
4447
4448
4449
4450 020732 005237 000304 TST147: INC @RSTESTN
4451 020736 012700 137777 MOV @137777,R0
4452 020742 076600 MED ;DISABLE WCS IF PRESENT
4453 020744 000352 352 ;BY DOING "INIT". XFC SHOULD
4454 020746 012700 021344 MOV @TABLE,TAB ;TRAP AA A RESERVED INSTR.
4455 020752 012002 GIN1: MOV (TAB)+,FIRST ;TABLE POINTER
4456 020754 012001 MOV (TAB)+,LAST ;FIRST OR CURRENT INSTRUCTION
4457 020756 020237 021400 CMP FIRST,FINISH ;LAST INSTRUCTION OF GROUP
4458 020762 001453 BEQ RET3A ;TESTED ALL
4459 020764 010237 021402 GIN2: MOV FIRST,INST ;GO TO MULT TRAPS TEST IF FIN.
4460 020770 012737 021012 000010 MOV @RET,RESVEC ;SET UP INST
4461 020776 012706 021774 MOV @BUFF,%6 ;LOAD RESERVED INST. TRAP VECTOR
4462 021002 005037 177776 CLR CC ;SET UP STACK POINTER
4463 021006 000137 021402 JMP INST ;CLEAR PRIORITY
4464
4465
4466
4467 021012 020627 021770 RET: CMP %6,@BUFF-4 ;TEST DECREMENT OF %6
4468 021016 001404 BEQ RET1
4469
4470 021020 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4471 021022 012713 000427 MOV #427,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4472 021026 000000 HLT ;WRONG DECREMENT
4473 021030 023727 021770 021404 RET1: CMP BUFF-4,@INST+2 ;LOC OF INST UNINCREMENTED
4474 021036 001405 BEQ RET2
4475
4476 021040 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4477 021042 012713 000430 MOV #430,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4478 021046 000000 HLT ;INST INC ON TRAP
4479 021050 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
4480
4481
4482 021052 005737 021772 RET2: TST BUFF-2 ;"JMP @RO" (000110) TO GET A
4483 021056 001405 BEQ RET3 ;SCOPE LOOP AND HIT CONTINUE
4484
4485 021060 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4486 021062 012713 000431 MOV #431,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4487 021066 000000 HLT ;CONDITION CODES SET ON TRAP
4488 021070 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
4489
4490
4491 021072 005237 021402 RET3: INC INST ;"JMP @RO" (000110) TO GET A
4492 021076 005202 INC FIRST ;SCOPE LOOP AND HIT CONTINUE
4493 021100 023701 021402 CMP INST,LAST
4494 021104 001722 BEQ GIN1 ;SET UP NEW GROUP
4495 021106 000137 020764 JMP GIN2 ;FINISH OLD GROUP
4496
4497 021112 010700 RET3A: SCOPE ;END OF INSTRUCTION GROUP
4498
4499
4500
4501

```

*TEST 150 CHECK MULTIPLE TRAP CONDITIONS

M07

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 90
 DOKDBA.P11 17-FEB-77 08:26 T150 CHECK MULTIPLE TRAP CONDITIONS

```

4502 021114 005237 000304 TST150: INC 2*STESTN
4503 ;THE FOLLOWING TESTS ARE WRITTEN TO CHECK MULTIPLE TRAP CONDITIONS
4504 ;TO TEST THAT TRAPS ARE PROCESSED PROPERLY.
4505 021120 012706 021774 SINSI: MOV #BUFF,%6 ;SET UP STACK POINTER
4506 021124 012737 021176 000014 MOV #SINSO,TBITVEC ;LOAD 'T' BIT VECTOR
4507 021132 012737 021222 000004 MOV #SINSE,4 ;LOAD ERROR VECTOR
4508 021140 012737 021205 000030 MOV #SINSF+1,EMTVEC ;LOAD EMT VECTOR (ODD)
4509 021146 005037 000032 CLR EMTVEC+2
4510 021152 005000 CLR %0 ;PRE SET R0
4511 021154 012746 000020 MOV #20,-(6) ;SET UP
4512 021160 012746 021166 MOV #SINSDD,-(6) ;TO SET 'T' BIT
4513 021164 000006 RTT ;SET 'T' BIT
4514 021166 104000 SINSDD: EMT ;DO EMT PICK UP ODD PC
4515
4516 021170 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4517 021172 012713 000432 SINSDD: MOV #432,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4518 021176 000000 HLT ;ERROR! 'T' TRAP OCCURRED BEFORE ERROR
4519 021200 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
4520 ;"JMP 2R0" (000110) TO GET A
4521 ;SCOPE LOOP AND HIT CONTINUE
4522 021202 000455 SINSF: BR SINSI ;GO TO SCOPE
4523 021204 000240 NOP ;THE EMT TRIES TO COME HERE
4524
4525 021206 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4526 021210 012713 000433 SINSDD: MOV #433,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4527 021214 000000 HLT ;ERROR! NO ERROR TRAP ON ODD EMT VECTOR
4528 021216 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
4529 ;"JMP 2R0" (000110) TO GET A
4530 ;SCOPE LOOP AND HIT CONTINUE
4531 021220 000446 SINSI: BR SINSI ;GO TO SCOPE
4532 021222 012737 021246 021764 SINSE: MOV #SINSG,BUFF-10 ;CORRECT BAD EMT VECTOR ON THE STACK
4533 021230 000006 RTT ;GO TO SINSG (SERVICE EMT)
4534
4535 021232 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4536 021234 012713 000434 SINSDD: MOV #434,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4537 021240 000000 HLT ;DIDN'T GO
4538 021242 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
4539 ;"JMP 2R0" (000110) TO GET A
4540 ;SCOPE LOOP AND HIT CONTINUE
4541 021244 000434 SINSI: BR SINSI ;GO TO SCOPE
4542 021246 012737 021276 021770 SINSG: MOV #SINSH,BUFF-4 ;RESTORE RETURN FROM EMT (ON STACK)
4543 021254 012737 021316 000014 MOV #SINSJ,TBITVEC ;CHANGE 'T' BIT VECTOR
4544 021262 000006 RTT ;GO TO SINSH
4545
4546 021264 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4547 021266 012713 000435 SINSDD: MOV #435,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4548 021272 000000 HLT ;REPLACE THIS INSTRUCTION BY A
4549 021274 000240 LOOP ;"JMP 2R0" (000110) TO GET A
4550 ;SCOPE LOOP AND HIT CONTINUE
4551
4552 021276 012700 000001 SINSH: MOV #1,%0 ;LOAD INDICATOR
4553
4554 021302 005212 INC (R2) ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
4555 021304 012713 000436 SINSDD: MOV #436,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
4556 021310 000000 HLT ;ERROR! SHOULD HAVE 'T' TRAPPED
4557 021312 000240 LOOP ;REPLACE THIS INSTRUCTION BY A
  
```

```

4558                                     ;"JMP 2RO" (000110) TO GET A
4559                                     ;SCOPE LOOP AND HIT CONTINUE
4560 021314 000410                                     BR      SINSI
4561 021316 022700 000001      SINSJ:  CMP      #1,%0
4562 021322 001405                                     BEQ     SINSI
4563
4564 021324 005212                                     INC     (R2)
4565 021326 012713 000437      MOV     #437,(R3)
4566 021332 000000                                     HLT
4567 021334 000240      LOOP
4568                                     ;SET MESSAGE TYPE (MSGTY) TO FATAL ERROR
4569                                     ;MOVE TO MAIL BOX (SFATAL) THE ERROR
4570 021336 010700      SINSI:  SCOPE
4571                                     ;ERROR! INST AT SINSH WAS NOT EXECUTED
4572 021340 000137 021420      JMP     ENDOJ
4573                                     ;REPLACE THIS INSTRUCTION BY A
4574                                     ;"JMP 2RO" (000110) TO GET A
4575                                     ;SCOPE LOOP AND HIT CONTINUE
4576 021344 000007      TABLE: 7
4577                                     ;END OF PROGRAM, GOT TO
4578                                     ;END OF TEST LINKAGE
4578 021346 000077                                     77
4579 021350 000210                                     210
4580 021352 000227                                     227
4581 021354 007000                                     7000
4582 021356 007777                                     7777
4583 021360 075040                                     75040
4584 021362 076577                                     76577
4585
4586 021364 076601                                     76601
4587 021366 076677                                     76677
4588
4589 021370 106400                                     106400
4590 021372 106477                                     106477
4591
4592 021374 106700                                     106700
4593 021376 107777                                     107777
4594 021400 021400      FINISH:  .
4595 021402 000000      INST:  HALT
4596 021404 000000                                     HALT
4597 021406 000000                                     HALT
4598 021410 000000                                     HALT
4599 021412 000000                                     HALT
4600 021414 000000      TEMP:  0
4601 021414 021420                                     .=. +2
4602
4603                                     ;END OF PASS LINKAGE
4604
4605
4606 021420 005237 000306      ENDOT:  INC     #SPASS
4607 021424 105237 021546      INCB   PASSPT
4608 021430 001027                                     BNE    ACT
4609 021432 132737 000040 000321      BITB  #40,SEVM
4610 021440 001023                                     BNE    ACT
4611 021442 023727 000042 021520      CMP   #42,#SENDAD
4612 021450 001417                                     BEQ    ACT
4613 021452 012700 021550      MOV   #MSG,RO
                                     ;GET MSG ADDR.
                                     ;END FLAG
                                     ;WILL CONTAIN RESERVED INST
                                     ;SHOULD TRAP TO LOC 10
                                     ;LOC 10 SHOULD SEND YOU TO
                                     ;RET
                                     ;SHOULD PRINT THIS PASS?
                                     ;NO
                                     ;WILL APT ALLOW PRINTING?
                                     ;NO

```

```

4614 021456 105737 177564      WAIT:  TSTB  @#TPS      ;TTY READY
4615 021462 100375              BPL  WAIT            ;NO WAIT
4616 021464 112037 177566      MOVB  (RO)+,@#TPB    ;PRINT CHARACTER
4617 021470 001372              BNE  WAIT            ;NEXT IF NOT DONE.
4618 021472 105737 177564      WAIT1: TSTB  @#TPS
4619 021476 100375              BPL  WAIT1
4620 021500 000005              RESET
4621 021502 012737 177761 021546  ACT:  MOV  #177761,PASSPT ;DO IT ABOUT 15 DECIMAL TIMES
4622 021510 013700 000042          MOV  @#42,RO          ;CHECK ACT
4623 021514 001405              BEQ  GOAGIN           ;KEEP GOING
4624 021516 007005              RESET
4625 021520 004710          SENDAD: JSR  PC,(RO)      ;ACT HOOKS
4626 021522 000240          NOP
4627 021524 000240          NOP
4628 021526 000240          NOP
4629 021530 012737 000012 000010  GOAGIN: MOV  #12,10
4630 021536 005037 000012          CLR  12
4631 021542 000137 001110          JMP  BEGIN1          ;DO NEXT PASS
4632 021546 177777          PASSPT: -1
4633 021550 005015 047105 020104  MSG:  .ASCIZ <15><12>/END OF PASS/
4634 021556 043117 020040 040520
4635 021564 051523 000
4636
4637
4638
4639
4640 021570 012737 021600 000024  PWRDWN: MOV  #PWRUP,24
4641 021576 000000          HALT
4642
4643 021600 012737 021570 000024  PWRUP:  MOV  #PWRDWN,24
4644 021606 012706 021774          MOV  #BUFF,SP
4645 021612 132737 000040 000321  BITB  #40,$ENVN      ;WILL APT ALLOW PRINTING?
4646 021620 001013          BNE  PFRES           ;NO
4647 021622 012700 021654          MOV  #MSGPWF,RO      ;GET MSG ADDR.
4648 021626 105737 177564      PWAIT: TSTB  @#TPS    ;TTY READY
4649 021632 100375          BPL  PWAIT           ;NO WAIT
4650 021634 112037 177566      MOVB  (RO)+,@#TPB    ;PRINT CHARACTER
4651 021640 001372          BNE  PWAIT           ;NEXT IF NOT DONE.
4652 021642 105737 177564      PWAIT1: TSTB  @#TPS
4653 021646 100375          BPL  PWAIT1
4654 021650 000137 001102  PFRES:  JMP  BEGIN
4655 021654 005015 047520 042527  MSGPWF: .ASCIZ <15><12>.POWER FAILED!.
4656 021662 020122 040506 046111
4657 021670 042105 000041
4658
4659
4660 021774 000000          .=.+100
4661 000001          BUFF:  0
          .END
  
```

ABASE = 000000	318			
ACOM1 = 000000	318			
ACOM2 = 000000	318			
ACPUOP = 000000	318	333		
ACT 021510	4608	4610	4612	4622#
ADDMD = 000000	318			
ADDW1 = 000000	318			
ADDW10 = 000000	318			
ADDW11 = 000000	318			
ADDW12 = 000000	318			
ADDW13 = 000000	318			
ADDW14 = 000000	318			
ADDW15 = 000000	318			
ADDW2 = 000000	318			
ADDW3 = 000000	318			
ADDW4 = 000000	318			
ADDW5 = 000000	318			
ADDW6 = 000000	318			
ADDW7 = 000000	318			
ADDW8 = 000000	318			
ADDW9 = 000000	318			
ADEVCT = 000000	318	324		
ADEVN = 000000	318			
RENV = 000000	318	329		
REVM = 000000	318	330		
RFATAL = 000000	318	321		
AIDR6 016616	3888#			
AMADR1 = 000000	318			
AMADR2 = 000000	318			
AMADR3 = 000000	318			
AMADR4 = 000000	318			
AMAMS1 = 000000	318			
AMAMS2 = 000000	318			
AMAMS3 = 000000	318			
AMAMS4 = 000000	318			
AMSGAD = 000000	318	326		
AMSLG = 000000	318	327		
AMSGTY = 000000	318	320		
AMTYP1 = 000000	318			
AMTYP2 = 000000	318			
AMTYP3 = 000000	318			
AMTYP4 = 000000	318			
APASS = 000000	318	323		
APRIOR = 000000	318			
ASWREG = 000000	318	331		
ATESTN = 000000	318	322		
AUNIT = 000000	318	325		
AUSWR = 000000	318	332		
AVECT1 = 000000	318			
AVECT2 = 000000	318			
BEGIN 001102	303	397#	4654	
BEGIN1 001110	401#	4631		
BELL = 000240	286#			
BGN2 001202	418#	419	421	
BGN2A 001216	423#	424		
BGN3 001224	413	416	425#	

CROSS REFERENCE TABLE -- USER SYMBOLS

BPT = 000003
BPTVEC= 000014
BUFF 021774

273#	2568												
279#	1017*	1018*	1034*	1053*	1072*	1088*	1109*	1110*	1127*	1128*	1144*		
408	432	449	452	469	472	487	491	504	508	526	545	571	
588	591	607	610	625	629	641	645	662	679	704	726	743	
746	762	765	781	785	797	801	818	836	861	878	881	897	
900	916	920	932	936	953	970	994	1016	1033	1036	1052	1055	
1071	1075	1087	1091	1108	1126	1153	1170	1173	1189	1192	1208	1212	
1224	1228	1245	1263	1286	1300	1303	1319	1322	1338	1342	1354	1358	
1375	1393	1417	1435	1438	1454	1457	1473	1477	1488	1492	1508	1525	
1546	1560	1562	1577	1595	1608	1625	1643	1682	1703	1734	1739	1753	
1767	1777	1794	1810	1823	1857	1863	1872	1887	1898	1914	1919	1934	
3670	3699	3712	3721	3730	3739	3754	3785	3796	3820	3839	3862	3888	
3891	3916	3918	3924	3932	3943	3946	3957	3970	3973	3984	4012	4013*	
4027	4028*	4352	4353*	4354*	4356	4367*	4368	4369*	4407*	4408	4409*	4461	
4467	4473	4482	4505	4532*	4542*	4644	4660#						
4045#													
4129#													

BXFR 017312
BYTOP 017640
CC = 177776

294#	489*	506*	529*	531	542*	548*	550	627*	643*	665*	667	676*	
682*	684	695*	783*	799*	821*	823	833*	839*	841	918*	934*	956*	
958	973*	975	1073*	1089*	1111*	1113	1123*	1129*	1131	1142*	1210*	1226*	
1248*	1250	1260*	1266*	1268	1279*	1340*	1356*	1378*	1380	1390*	1396*	1398	
1409*	1475*	1490*	1511*	1513	1522*	1529	1539*	1931*	2341*	2814*	3223*	3322*	
3341	3416*	3694*	3755*	4462*									
277#	812*	863*	879*	898*	917*	933*	954*	955*	971*	972*	1008*	1009*	
2331#	2332*	2948*	2949*	3013*	3216*	3217*	3305*	3306*	4508*	4509*			
4572	4506#												
275#	1154*	1155*	1171*	1190*	1209*	1225*	1246*	1247*	1264*	1265*	1287*	1288*	
1301*	1320*	1339*	1355*	1376*	1377*	1394*	1395*	1418*	1419*	1436*	1455*	1474*	
1489*	1517*	1510*	1526*	1527*	1694*	1695*	1843*	1844*	1862*	1879*	1896*	1910*	
1930*	2334*	2340*	2565*	2566*	2616*								
4457	4594#												

EMTVEC= 000030

ENDOT 021420
ERRVEC= 000004

FINISH 021400
FOVER 010500
FOV1 010504
GIN1 020752
GIN2 020764
GORGIN 021530
HLT = 000000

271#	439	457	477	496	514	536	556	578	596	615	634	651	
672	690	709	733	751	770	790	807	828	847	868	886	905	
925	942	963	981	999	1023	1041	1060	1080	1097	1118	1137	1160	
1178	1197	1217	1234	1255	1274	1293	1308	1327	1347	1364	1385	1404	
1424	1443	1462	1482	1498	1518	1535	1554	1567	1585	1602	1615	1633	
1655	1664	1672	1688	1714	1723	1744	1762	1772	1782	1806	1815	1825	
1850	1868	1884	1902	1916	1936	1946	1970	1974	1991	1995	2011	2015	
2032	2036	2053	2057	2074	2078	2088	2105	2109	2134	2143	2163	2172	
2192	2201	2221	2230	2250	2259	2299	2347	2357	2367	2377	2386	2397	
2407	2417	2427	2437	2447	2457	2467	2477	2487	2497	2507	2517	2527	
2537	2547	2577	2586	2596	2605	2627	2636	2646	2655	2670	2689	2706	
2716	2725	2734	2743	2762	2771	2780	2789	2798	2821	2827	2836	2845	
2857	2866	2884	2901	2919	2937	2960	2967	2976	2991	3000	3009	3033	
3040	3049	3063	3072	3081	3104	3111	3120	3134	3143	3166	3173	3182	
3194	3203	3229	3237	3247	3256	3265	3274	3290	3299	3328	3336	3346	
3355	3364	3373	3389	3398	3422	3429	3438	3447	3456	3465	3481	3490	
3509	3516	3533	3540	3546	3564	3571	3577	3594	3603	3622	3629	3635	
3658	3683	3707	3717	3726	3735	3744	3764	3772	3807	3833	3851	3874	
3896	3910	3923	3937	3951	3965	3978	3992	4005	4021	4036	4055	4071	
4087	4103	4119	4134	4145	4156	4167	4178	4189	4200	4211	4230	4244	

		4258	4271	4283	4291	4299	4307	4320	4328	4336	4344	4361	4377	4385
		4393	4401	4416	4424	4432	4440	4472	4478	4487	4518	4527	4537	4548
		4556	4566											
ILLA =	004700	292#	2070											
ILLB =	000100	293#	2101											
INDEX	011754	2682#												
IND1	012010	2683	2693#											
IND10	013006	2930	2937#											
IND11	012742	2915	2923#											
IND12	013012	2933	2941#											
IND2	012064	2697	2711#											
IND2A	012056	2706#												
IND2B	012050	2702#	2720											
IND2X	012170	2710	2739	2747#										
IND3	012236	2754	2766#											
IND3A	012232	2762#												
IND3B	012224	2758#	2775											
IND4	012436	2809	2822#											
IND4A	012434	2821#												
IND4B	012426	2817#	2852											
IND5	012626	2877	2884#											
IND6	012670	2895	2901#											
IND7	012736	2912	2919#											
IND8	012632	2880	2888#											
IND9	012674	2897	2905#											
INST	021402	4459#	4463	4473	4491#	4493	4595#							
INSTC	001346	471#												
INSTK	004730	1321#	1322											
INT	015770	3696	3703#	3721	3739									
INT1	016116	3740	3748#											
INT2	016002	3698	3711#											
IOTVEC=	000020	280#	727*	728*	744*	763*	782*	798*	819*	820*	837*	838*	854*	855*
		2335#	2336*	3093*	3094*	3147*								
		274#												
ITRAPS=	000004	372#	4014*	4015*	4016	4029*	4030*	4031	4046*	4047	4050	4062*	4063	4066
K1	001000	4077#	4079	4093*	4095	4109*	4111	4114						
		379#	4151	4162	4173	4184	4195	4206						
K10	001016	380#	4222*	4224*	4225	4236	4237*	4239	4250	4251*	4253	4264	4266	
K11	001020	381#												
K12	001022	373#												
K2	001002	374#												
K3	001004	375#												
K4	001006	376#	4045*	4048	4061*	4078*	4080	4082	4094*	4096	4098	4110*	4112	
K5	001010	377#	4064	4162										
K6	001012	378#	4129	4140	4151	4173	4206							
K7	001014	287#	440	460	479	499	517	539	559	579	598	617	636	653
LOOP =	000240	673	692	711	734	753	772	792	809	830	849	869	888	907
		927	944	965	983	1001	1024	1043	1062	1082	1099	1120	1139	1161
		1180	1199	1219	1236	1257	1276	1294	1310	1329	1349	1366	1387	1406
		1425	1445	1464	1483	1499	1519	1536	1555	1568	1586	1603	1616	1634
		1656	1665	1673	1689	1715	1726	1745	1764	1774	1784	1807	1816	1828
		1852	1869	1885	1904	1918	1937	1950	1975	1996	2016	2037	2058	2079
		2089	2110	2135	2147	2164	2176	2193	2205	2222	2234	2251	2263	2303
		2348	2358	2368	2378	2388	2399	2409	2419	2429	2439	2449	2459	2469
		2479	2489	2499	2509	2519	2529	2539	2550	2578	2587	2597	2606	2628
		2637	2647	2656	2671	2690	2707	2717	2726	2735	2744	2763	2772	2781

RETC4	003654	1053	1055#												
RETC5	004312	1190	1192#												
RETD	001420	488	491#												
RETD1	002052	626	629#												
RETD2	002566	782	785#												
RETD3	003220	917	920#												
RETD4	003724	1072	1075#												
RETD5	004362	1209	1212#												
RETE	001464	505	508#												
RETE1	002116	642	645#												
RETE2	002632	798	801#												
RETE3	003264	933	936#												
RETE4	003770	1088	1091#												
RETE5	004426	1225	1228#												
RETF	001544	527	531#												
RETF1	002176	663	667#												
RETF2	002712	819	823#												
RETF3	003344	954	958#												
RETF4	004050	1109	1113#												
RETF5	004506	1246	1250#												
RETG	001620	546	550#												
RETG1	002252	680	684#												
RETG2	002766	837	841#												
RETG3	003414	971	975#												
RETG4	004024	1127	1131#												
RETG5	004562	1264	1268#												
RETH5	004652	1287	1297#												
RETJ	004670	1301	1303#												
RETK	004732	1320	1322#												
RETL	005002	1339	1342#												
RETM	005046	1355	1358#												
RETN	005126	1376	1380#												
RETO	005202	1394	1398#												
RETP	005274	1418	1428#												
RETO	005320	1436	1438#												
RETR	005364	1455	1457#												
RETS	005436	1474	1477#												
RETT	005504	1489	1492#												
RETV	005566	1509	1513#												
RETV	005640	1526	1529#												
RET1	021030	4468	4473#												
RET2	021052	4474	4482#												
RET3	021072	4483	4491#												
RET3A	021112	4458	4497#												
RL0BYT	020104	4222#													
RL0BY1	020732	4445#													
R7TR1	015214	3528	3541#												
R7TR2	015302	3558	3572#												
R7TR2A	015260	3560#	3572												
R7TR3	015360	3588	3598#												
R7TR3A	015346	3590#	3598												
R7TR4	015450	3615	3630#												
SCOPE =	010700	270#	427	443	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	

TR3	015664	3673	3679#
TR4	015700	3675	3687#
TR4A	015426	3618#	3630
TR5	015674	3676	3683#
TR6	016206	3756	3772#
TR6X	016214	3768	3778#
TST1	001226	431#	
TST10	001762	606#	
TST100	010030	2126#	
TST101	010106	2155#	
TST102	010164	2184#	
TST103	010242	2213#	
TST104	010320	2242#	
TST105	010376	2273#	
TST106	010506	2311#	
TST107	011332	2560#	
TST11	002026	624#	
TST110	011750	2681#	
TST111	012172	2752#	
TST112	012344	2807#	
TST113	012564	2875#	
TST114	012634	2893#	
TST115	012676	2910#	
TST116	012744	2928#	
TST117	013014	2946#	
TST12	002144	661#	
TST120	013232	3019#	
TST121	013446	3091#	
TST122	013632	3153#	
TST123	014014	3213#	
TST124	014316	3312#	
TST125	014600	3407#	
TST126	015052	3499#	
TST127	015150	3526#	
TST13	002304	701#	
TST130	015240	3556#	
TST131	015326	3586#	
TST132	015404	3613#	
TST133	015502	3646#	
TST134	015602	3669#	
TST135	015712	3693#	
TST136	016120	3753#	
TST137	016222	3784#	
TST14	002374	725#	
TST140	016264	3795#	
TST141	016350	3818#	
TST142	016526	3860#	
TST143	016612	3886#	
TST144	017306	4044#	
TST145	017634	4128#	
TST146	020100	4221#	
TST147	020732	4450#	
TST15	002434	742#	
TST150	021114	4502#	
TST16	002476	761#	
TST17	002542	780#	

TTY3	016406	3821	3826																
TTY4	016472	3840	3844																
VDEC1	017410	1965	1978																
VDEC10	007654	2047	2057																
VDEC11	007732	2069	2082																
VDEC12	007726	2068	2078																
VDEC13	010026	2100	2113																
VDEC14	010022	2099	2109																
VDEC15	010064	2129	2138																
VDEC16	010142	2158	2167																
VDEC17	010220	2187	2196																
VDEC18	010276	2216	2225																
VDEC19	010354	2245	2254																
VDEC2	007404	1964	1974																
VDEC3	007462	1986	1999																
VDEC4	007456	1985	1995																
VDEC5	017534	2006	2019																
VDEC6	007530	2005	2015																
VDEC7	007606	2027	2040																
VDEC8	007602	2026	2036																
VDEC9	007660	2048	2061																
WAIT	021456	4614	4615	4617															
WAIT1	021472	4618	4619																
SAPTHD	000330	351	357																
SCPUOP	000326	333																	
SEVCT	000310	324																	
SENDAD	021520	311	4611	4625															
SENO =	000440	267	436	439	454	457	474	477	493	496	511	514	533	536					
		553	556	575	578	593	596	612	615	631	634	648	651	669					
		672	687	690	706	709	730	733	748	751	767	770	787	790					
		804	807	825	828	844	847	865	868	883	886	902	905	922					
		925	939	942	960	963	978	981	996	999	1020	1023	1038	1041					
		1057	1060	1077	1080	1094	1097	1115	1118	1134	1137	1157	1160	1175					
		1178	1194	1197	1214	1217	1231	1234	1252	1255	1271	1274	1290	1293					
		1305	1308	1324	1327	1344	1347	1361	1364	1382	1385	1401	1404	1421					
		1424	1440	1443	1459	1462	1479	1482	1495	1498	1515	1518	1532	1535					
		1551	1554	1564	1567	1582	1585	1599	1602	1612	1615	1630	1633	1652					
		1655	1661	1664	1669	1672	1685	1688	1711	1714	1720	1723	1741	1744					
		1759	1762	1769	1772	1779	1782	1803	1806	1812	1815	1822	1825	1847					
		1850	1865	183	1881	1884	1899	1902	1913	1916	1933	1936	1943	1946					
		1967	1970	1971	1974	1908	1991	1992	1995	2008	2011	2012	2015	2029					
		2032	2033	2036	2050	2053	2054	2057	2071	2074	2075	2078	2085	2088					
		2102	2105	2106	2109	2131	2134	2140	2143	2160	2163	2169	2172	2189					
		2192	2198	2201	2218	2221	2227	2230	2247	2250	2256	2259	2296	2299					
		2344	2347	2354	2357	2364	2367	2374	2377	2383	2386	2394	2397	2404					
		2407	2414	2417	2424	2427	2434	2437	2444	2447	2454	2457	2464	2467					
		2474	2477	2484	2487	2494	2497	2504	2507	2514	2517	2524	2527	2534					
		2537	2544	2547	2574	2577	2583	2586	2593	2596	2602	2605	2624	2627					
		2633	2636	2643	2646	2652	2655	2667	2670	2686	2689	2703	2706	2713					
		2716	2722	2725	2731	2734	2740	2743	2759	2762	2768	2771	2777	2780					
		2786	2789	2795	2798	2818	2821	2824	2827	2833	2836	2842	2845	2854					
		2857	2863	2866	2881	2884	2898	2901	2916	2919	2934	2937	2957	2960					
		2964	2967	2973	2976	2988	2991	2997	3000	3006	3009	3030	3033	3037					
		3040	3046	3049	3060	3063	3069	3072	3078	3081	3101	3104	3108	3111					
		3117	3120	3131	3134	3140	3143	3163	3166	3170	3173	3179	3182	3191					
		3194	3200	3203	3226	3229	3234	3237	3244	3247	3253	3256	3262	3265					

M08

3271	3274	3287	3290	3296	3299	3325	3328	3333	3336	3343	3346	3352
3355	3361	3364	3370	3373	3386	3389	3395	3398	3419	3422	3426	3429
3435	3438	3444	3447	3453	3456	3462	3465	3478	3481	3487	3490	3506
3509	3513	3516	3530	3533	3537	3540	3543	3546	3561	3564	3568	3571
3574	3577	3591	3594	3600	3603	3619	3622	3626	3629	3632	3635	3655
3658	3680	3683	3704	3707	3714	3717	3723	3726	3732	3735	3741	3744
3761	3764	3769	3772	3804	3807	3830	3833	3848	3851	3871	3874	3893
3896	3907	3910	3920	3923	3934	3937	3948	3951	3962	3965	3975	3978
3989	3992	4002	4005	4018	4021	4033	4036	4052	4055	4068	4071	4084
4087	4100	4103	4116	4119	4131	4134	4142	4145	4153	4156	4164	4167
4175	4178	4186	4189	4197	4200	4208	4211	4227	4230	4241	4244	4255
4258	4268	4271	4280	4283	4288	4291	4296	4299	4304	4307	4317	4320
4325	4328	4333	4336	4341	4344	4358	4361	4374	4377	4382	4385	4390
4393	4398	4401	4413	4416	4421	4424	4429	4432	4437	4440	4469	4472
4475	4478	4484	4487	4515	4518	4524	4527	4534	4537	4545	4548	4553
4556	4563	4566										
329	415	4609	4645									
330	405*											
366												
328												
340	363											
321	366	407										
358												
319	359	363										
359												
326												
327												
320	401*	406										
323	4606*											
361												
268	428	432	445	449	465	469	483	487	522	526	567	571
584	588	603	607	621	625	658	662	698	702	722	726	739
743	758	762	777	781	814	818	857	861	874	878	893	897
912	916	949	953	988	992	1012	1016	1029	1033	1048	1052	1067
1071	1104	1108	1149	1153	1166	1170	1185	1189	1204	1208	1241	1245
1282	1286	1315	1319	1334	1338	1371	1375	1413	1417	1431	1435	1450
1454	1469	1473	1504	1508	1542	1546	1573	1577	1591	1595	1621	1625
1639	1643	1678	1682	1699	1703	1730	1734	1749	1753	1790	1794	1838
1842	1857	1861	1874	1878	1890	1894	1923	1927	1959	1963	1980	1984
2000	2004	2021	2025	2042	2046	2063	2067	2094	2098	2123	2127	2152
2156	2181	2185	2210	2214	2239	2243	2270	2274	2308	2312	2557	2561
2678	2682	2749	2753	2804	2808	2872	2876	2890	2894	2907	2911	2925
2929	2943	2947	3016	3020	3088	3092	3150	3154	3210	3214	3309	3313
3404	3408	3496	3500	3523	3527	3553	3557	3583	3587	3610	3614	3643
3647	3666	3670	3690	3694	3750	3754	3781	3785	3792	3796	3815	3819
3857	3861	3883	3887	4041	4045	4125	4129	4218	4222	4447	4451	4499
4503												
309	314											
331												
322	365	431*	448*	468*	486*	525*	570*	587*	606*	624*	661*	701*
725	742*	761*	780*	817*	860*	877*	896*	915*	952*	991*	1015*	1032*
1051	1070*	1107*	1152*	1169*	1188*	1207*	1244*	1285*	1318*	1337*	1374*	1416*
1434	1453*	1472*	1507*	1545*	1576*	1594*	1624*	1642*	1681*	1702*	1733*	1752*
1793	1841*	1860*	1877*	1893*	1926*	1962*	1983*	2003*	2024*	2045*	2066*	2097*
2126	2155*	2184*	2213*	2242*	2273*	2311*	2560*	2681*	2752*	2807*	2875*	2893*
2910	2928*	2946*	3019*	3091*	3153*	3213*	3312*	3407*	3499*	3526*	3556*	3586*

SENV 000320
 SENVM 000321
 SEERROR= 000302
 SETABL 000320
 SETEND 000330
 SFATAL 000302
 SHIBTS 000330
 SMAIL 000300
 SOR 000332
 SLOAD 000314
 SGLG 000316
 SACTY 000300
 SPASS 000306
 SPASTM 000336
 SSN = 000151

SSVPC = 000300
 SSWREG 000322
 STESTM 000304

AEROR	258#	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	1038	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	1669	1685	1711	1720	1741	1759	1769	1779
	1803	1812	1822	1847	1865	1881	1899	1913	1933	1943	1967	1971	1988	1992	2008
	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	2898	2916	2934	2957	2964	2973
	2988	2997	3006	3030	3037	3046	3060	3069	3078	3101	3108	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	3478	3487	3506	3513
	3530	3537	3543	3561	3568	3574	3591	3600	3619	3626	3632	3655	3680	3704	3714
	3723	3732	3741	3761	3769	3804	3830	3848	3871	3893	3907	3920	3934	3948	3962
	3975	3989	4002	4018	4033	4052	4068	4084	4100	4116	4131	4142	4153	4164	4175
	4186	4197	4208	4227	4241	4255	4268	4280	4288	4296	4304	4317	4325	4333	4341
	4358	4374	4382	4390	4398	4413	4421	4429	4437	4469	4475	4484	4515	4524	4534
	4545	4553	4563												
APTERO	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	1038	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	1669	1685	1711	1720	1741	1759	1769	1779
	1803	1812	1822	1847	1865	1881	1899	1913	1933	1943	1967	1971	1988	1992	2008
	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	2898	2916	2934	2957	2964	2973
	2988	2997	3006	3030	3037	3046	3060	3069	3078	3101	3108	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	3478	3487	3506	3513
	3530	3537	3543	3561	3568	3574	3591	3600	3619	3626	3632	3655	3680	3704	3714
	3723	3732	3741	3761	3769	3804	3830	3848	3871	3893	3907	3920	3934	3948	3962
	3975	3989	4002	4018	4033	4052	4068	4084	4100	4116	4131	4142	4153	4164	4175
	4186	4197	4208	4227	4241	4255	4268	4280	4288	4296	4304	4317	4325	4333	4341
	4358	4374	4382	4390	4398	4413	4421	4429	4437	4469	4475	4484	4515	4524	4534
	4545	4553	4563												
HDR	261#	428	445	465	483	522	567	584	603	621	658	698	722	739	758
	777	814	857	874	893	912	949	988	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	1838	1857	1874	1890	1923	1959	1980	2000
	2021	2042	2063	2094	2123	2152	2181	2210	2239	2270	2308	2557	2678	2749	2804
	2872	2890	2907	2925	2943	3016	3088	3150	3210	3309	3404	3496	3523	3553	3583
	3610	3643	3666	3690	3750	3781	3792	3815	3857	3883	4041	4125	4218	4447	4499
SETTBI	259#	1707	1736	1755	3700	3799									
STARS	258#	307	317	344	346	353									
THOR	264#	428	445	465	483	522	567	584	603	621	658	698	722	739	758
	777	814	857	874	893	912	949	988	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	1838	1857	1874	1890	1923	1959	1980	2000
	2021	2042	2063	2094	2123	2152	2181	2210	2239	2270	2308	2557	2678	2749	2804
	2872	2890	2907	2925	2943	3016	3088	3150	3210	3309	3404	3496	3523	3553	3583
	3610	3643	3666	3690	3750	3781	3792	3815	3857	3883	4041	4125	4218	4447	4499

.MAIN. MACY11 27(1006) 17-FEB-77 08:32 PAGE 108
DQKDBA.P11 17-FEB-77 08:26 CROSS REFERENCE TABLE -- MACRO NAMES

VTRPO	2115#	2123	2152	2181	2210	2239
.SACT1	300#	305				
.SAPT8	300#	315				
.SAPTH	300#	342				

. ABS. 021776 000

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

DSKZ:DQKDBA, DSKZ:DQKDBA.SEQ/CRF/SOL=DSKZ:DQKDBA.P11
RUN-TIME: 14 12 1 SECONDS
RUN-TIME RATIO: 423/28=14.9
CORE USED: 10K (20 PAGES)